University of Windsor

Scholarship at UWindsor

Electronic Theses and Dissertations

Theses, Dissertations, and Major Papers

1-1-2007

Neural network-based shape retrieval using fuzzy clustering and moment-based representations.

Nan Xing

University of Windsor

Follow this and additional works at: https://scholar.uwindsor.ca/etd

Recommended Citation

Xing, Nan, "Neural network-based shape retrieval using fuzzy clustering and moment-based representations." (2007). *Electronic Theses and Dissertations*. 7124.
https://scholar.uwindsor.ca/etd/7124

This online database contains the full-text of PhD dissertations and Masters' theses of University of Windsor students from 1954 forward. These documents are made available for personal study and research purposes only, in accordance with the Canadian Copyright Act and the Creative Commons license—CC BY-NC-ND (Attribution, Non-Commercial, No Derivative Works). Under this license, works must always be attributed to the copyright holder (original author), cannot be used for any commercial purposes, and may not be altered. Any other use would require the permission of the copyright holder. Students may inquire about withdrawing their dissertation and/or thesis from this database. For additional inquiries, please contact the repository administrator via email (scholarship@uwindsor.ca) or by telephone at 519-253-3000ext. 3208.

Neural Network-Based Shape Retrieval Using Fuzzy Clustering and Moment-Based Representations

by

Nan Xing

A Thesis

Submitted to the Faculty of Graduate Studies and Research

through Computer Science

In Partial Fulfillment of the Requirements for

the Degree of Master of Science at the

University of Windsor

Windsor, Ontario, Canada

2007

© 2007 Nan Xing

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Library and
Archives Canada

Bibliothèque et
Archives Canada

Published Heritage
Branch

Direction du
Patrimoine de l'édition

395 Wellington Street
Ottawa ON K1A 0N4
Canada

395, rue Wellington
Ottawa ON K1A 0N4
Canada

Your file Votre référence
ISBN: 978-0-494-42315-8
Our file Notre référence
ISBN: 978-0-494-42315-8

NOTICE:
The author has granted a non-exclusive license allowing Library and Archives Canada to reproduce, publish, archive, preserve, conserve, communicate to the public by telecommunication or on the Internet, loan, distribute and sell theses worldwide, for commercial or non-commercial purposes, in microform, paper, electronic and/or any other formats.

The author retains copyright ownership and moral rights in this thesis. Neither the thesis nor substantial extracts from it may be printed or otherwise reproduced without the author's permission.

AVIS:
L'auteur a accordé une licence non exclusive permettant à la Bibliothèque et Archives Canada de reproduire, publier, archiver, sauvegarder, conserver, transmettre au public par télécommunication ou par l'Internet, prêter, distribuer et vendre des thèses partout dans le monde, à des fins commerciales ou autres, sur support microforme, papier, électronique et/ou autres formats.

L'auteur conserve la propriété du droit d'auteur et des droits moraux qui protège cette thèse. Ni la thèse ni des extraits substantiels de celle-ci ne doivent être imprimés ou autrement reproduits sans son autorisation.

In compliance with the Canadian Privacy Act some supporting forms may have been removed from this thesis.

While these forms may be included in the document page count, their removal does not represent any loss of content from the thesis.

Conformément à la loi canadienne sur la protection de la vie privée, quelques formulaires secondaires ont été enlevés de cette thèse.

Bien que ces formulaires aient inclus dans la pagination, il n'y aura aucun contenu manquant.

Canada

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

ABSTRACT

The shape of an object is a fundamental image feature and belongs to one of the most important image features used in Content-Based Image Retrieval (CBIR). In this thesis, we propose use of Neural Network-Based Shape Retrieval System in which Moment Invariants and Zernike Moments serve as the feature vector to describe shape. Fuzzy *K*-means Clustering is used to group similar images in an image collection into *K*-clusters whereas Neural Network is used to facilitate retrieval against a given query image. Neural Network is trained by the clustering results of all of the images in the collection in which feature vector formed by moments serves as its inputs and the output dictates the degree of membership among the *K*-Clusters. The suggested approach is compared with *K*-means Clustering for a number of different distance functions.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Acknowledgements

First and foremost, I really appreciate my academic advisor, Dr. Imran Ahmad. Without his constant guidance, I would not have been able to finish my work. His knowledge, patience and serious attitude made me learn a lot throughout my graduate program.

I would also like to acknowledge Dr. Alioune Ngom, Dr. Jonathan Wu, and Dr. Dan Wu for spending their precious time in reading my thesis and providing their valuable comments and advice on my work.

My special thanks go to my parents. It would not have been possible to finish my work without your support.

Thanks to my friends, I really appreciate your support.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Table of Contents

Abstract iii

Acknowledgements iv

List of Figures vii

1. Introduction 1
    - 1.1 Text-Based Image Retrieval 1
    - 1.2 Content-Based Image Retrieval 2
    - 1.3 Visual Features 3
        - 1.3.1 Color 3
        - 1.3.2 Texture 4
        - 1.3.3 Shape 4
    - 1.4 Problem Statement 5

2. Background 7
    - 2.1 Moments 7
        - 2.1.1 Moment Invariants 7
        - 2.1.2 Zernike Moments 8
    - 2.2 Neural Networks 9
        - 2.2.1 Multilayer Neural Networks 11
    - 2.3 Clustering 13
        - 2.3.1 K-means Clustering 14
        - 2.3.2 Fuzzy K-means Clustering 15
        - 2.3.3 Distance Functions 16
    - 2.4 User Feedback 16

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

3. Methodology 18
3.1 System Architecture 18
3.2 Feature Extraction 19
3.3 Fuzzy K-means Clustering 22
3.4 Neural Network 24
3.5 Distance Functions 26

4. Experiments and Discussions 30
4.1 Development Environment and GUI 30
4.2 Image Collection 31
4.3 Performance Measurement 32
4.4 Comparison of Distance Functions 33
4.5 Comparison of Fuzzy K-means Clustering and K-means Clustering 35
4.6 Comparison of number of clusters in Fuzzy K-means Clustering 42
4.7 Comparison of the threshold for retrieval 44
4.8 Evaluation of System Performance and the Training Samples for Moment Invariants 46
4.9 Determination of the number of relevant images in a cluster 48
4.10 Comparison of Image Sizes 49
5. Conclusion and Future Works 52
5.1 Conclusion 52
5.2 Future Directions 52

References 53

Vita Auctoris 60

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

List of Figures

Figure 1 An image taken in Toronto 2
Figure 2 TLU 10
Figure 3 A Multilayer (three-layer) Neural Network 11
Figure 4 Types of decision regions that can be formed by single-layer and multi-layer neural networks 12
Figure 5 The dots grouped into 4 clusters 13
Figure 6 Subjectivity of human perception 17
Figure 7 System Architecture 19
Figure 8 Fuzzy K-means Clustering 23
Figure 9 Data with unequal coordinate variances 28
Figure 10 Data with equal coordinate variances 29
Figure 11 GUI and a sample retrieval action 30
Figure 12 Sample Binary Images in the Image Collection 31
Figure 13 Comparison of Distance Function for Moment Invariants 33
Figure 14 Comparison of Distance Function for Zernike Moments 34
Figure 15 Comparison of Fuzzy K-means Clustering and K-means Clustering for Moment Invariants using Correlation Distance 35
Figure 16 Comparison of Fuzzy K-means Clustering and K-means Clustering for Zernike Moments using Correlation Distance 37
Figure 17 Comparison of Fuzzy K-means Clustering and K-means Clustering for Moment Invariants using Euclidean Distance 39
Figure 18 Comparison of Fuzzy K-means Clustering and K-means Clustering for Zernike Moments using Euclidean Distance 40
Figure 19 Comparison of Number of Clusters in Fuzzy K-means Clustering for Moment Invariants 42

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Figure 20 Comparison of Number of Clusters in Fuzzy K-means Clustering for Zernike Moments ... 43
Figure 21 Comparison of the threshold for retrieval for Moment Invariants ... 44
Figure 22 Comparison of the threshold for retrieval for Zernike Moments ... 45
Figure 23 Evaluation of System Performance and the Training Samples for Moment Invariants ... 46
Figure 24 Evaluation of System Performance and the Training Samples for Zernike Moments ... 47
Figure 25 Determination of the number of relevant images in a cluster for Moment Invariants ... 48
Figure 26 Determination of the number of relevant images in a cluster for Zernike Moments ... 48
Figure 27 Comparison of Image Sizes for Moment Invariants ... 50
Figure 28 Comparison of Image Sizes for Zernike Moments ... 50

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

1. Introduction

Rapid technological advances and digital imaging revolution has resulted in generation and storage of a large number of images. A number of application domains are utilizing such image collections for everyday use. Examples of such applications can be found in everyday life, from entrainment to medicine, from sports to training, where million of images are generated and growing every year. In order to utilize these image collections effectively, there is need to search for desired images both quickly and efficiently. However, effective and efficient search through these large image collections poses significant technical challenges. Image retrieval is a technique to find similar images from a given image collection on the basis of the similarity of image contents.

1.1 Text-Based Image Retrieval

Earlier image retrieval approaches were all text-based and can be traced back to early 1970's [3]. In this approach, keywords and textual annotation are associated with various salient image features. Traditional Database Management System (DBMS) techniques are used to store, index, search, and retrieve relevant images from the image collection against a given query image on the basis of matching keywords and annotations. After all the images are stored in the database, one can make a query using these keywords and textual annotation to find a specified image through the DBMS.

Though this idea is straightforward and easy to implement, it suffers from many disadvantages. Most important among them are the manual labor involved in the identification and annotation of salient image features and the subjectivity of human perception. Manual annotation of image contents is a tedious process. If the image collection is very large, it will become awfully difficult, exhaustive and time consuming to annotate all of the images. Moreover, this will suffer from the problem of human subjectivity since different people may

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

perceive the same image differently.

Figure 1: An image taken in Toronto

As an example, consider the images given in Figure 1. It is an image of the Toronto skyline, featuring CN tower. Among others, this image may be annotated as 'Toronto' or 'CN Tower' or both. If the image is annotated as 'CN Tower' and user issues a query for 'Toronto' or vice versa, system will not be able to find this image even though the image does exist in the database.

1.2 Content-Based Image Retrieval (CBIR)

The problems mentioned above were significant to generate the idea of Content-Based Image Retrieval (CBIR). In this approach, instead of annotating the images manually, images are indexed by their contents. The main objective of content-based image retrieval is the automatic extraction of salient visual image features for determination of similarity among the database images and a given query image. All of these approaches are based on extraction of quantifiable low-level visual image features such as color, texture, shape, spatial positions, etc.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

These image features are generally described by multi-dimensional feature vectors. For retrieval, users provide the system with example images, from which system then extracts internal representation of feature vectors. The similarity or distance between the feature vectors of the query image and those of the images in the database are then calculated and retrieval is performed only with the aid of an indexing scheme.

1.3 Visual Features

Visual features of an image can be classified as general features and domain-specific features. General features include color, texture and shape while domain-specific features may include features specific to that domain such as human face for face database applications, finger prints for finger print matching, etc. Following sections provide a brief review of the most common general image features.

1.3.1 Color

Color is one of the most commonly used features in CBIR. We often use Color Histogram as the representation of this feature. From the perspective of statistics, color histogram represents the joint probability of three color channels which, most commonly, are red, green and blue. There are also other feature representations for color. In [4], the Color Moments method is proposed, which is based on the truth that the color distribution can be characterized by the color moments. Another approach as proposed in [44] is called Color Sets. In this approach, the RGB color space is first transformed into a perceptually uniform space, such as HSV space and then transformed further into some bins of colors which are called color sets.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

1.3.2 Texture

In [6], texture is defined as the visual patterns that have the properties of homogeneity that do not result from the presence of only a single color or intensity. For instance, the fire, cloud, grass, sea etc are the best representatives of texture. Instead of single pixel, texture does not only present the information of the pixel itself, but it also stores the structural information and the relationship to the surrounding area. The approach in texture representation such as co-occurrence matrix representation has a long history. In 1970's, Haralick proposed the co-occurrence matrix representation. The Wavelet transform may be the most famous technique in texture representation when it was proposed in 1990's [7]. This method used Wavelet signal to transform the image from time domain to frequency domain and got good results.

1.3.3 Shape

The shape of an object is a fundamental image feature and belongs to one of the most important image features used in Content-Based Image Retrieval (CBIR). Shape description is one of the key parts of image content description for shape-based image retrieval.

For many of the image processing and computer vision applications, shape of an object can be described through its binary representation. In this representation, the shape area is described by a single color whereas rest is defined as only the background, represented in a different color. As an example, we use black color to represent the pixel inside the object area while we use white color to represent the pixel outside the object area. There are two main categories of approaches for shape representation:

(i) Boundary-based approach: in this approach, shape of an object is represented with help of its boundary.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

(ii) Region-based approach which focuses on the whole region of the object in the image.

Regardless of the method for shape representation, [8] provides a set of criterion that a shape representation scheme must provide:

- Unique: each shape should have a unique representation.
- Compact: a good representation should be compact to save storage space.
- Accurate and Reliable: a good representation should reflect the shape features accurately and be robust.
- Invariant: a shape representation should be invariant to geometric transformations such as rotation, scaling and translation.

1.4 Problem Statement

This thesis is an addition to an existing Shape-Based Image Retrieval System in which Moment Invariants and Zernike Moments are used as the feature vectors. In that system, Neural Network is used as the main classifier whereas K-means clustering is used to group similar images in non-overlapping *K* distinct clusters.

This thesis investigates use of Fuzzy K-means Clustering for image classification and compares its performance against the existing scheme that involved K-means Clustering with K-means Clustering. In the earlier proposed system, Euclidean and Mahalanobis functions were used as distance functions to evaluate system performance. In addition to these two distance functions, in this thesis, Correlation Distance and Standardized Euclidean Distance functions are used to evaluate and compare system performance.

The rest of the thesis is organized as follows: the background of this thesis is

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

discussed in Chapter 2. Chapter 3 discusses our proposed methodology. Experimental results are presented in Chapter 4. Chapter 5 provides our conclusions and future work directions.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

2. Background

2.1 Moments

2.1.1 Moment Invariants

Moments are widely used in visual information processing. Moment Invariants are derived from shape moments and are not changed during two-dimensional geometric transformations such as rotation, translation and scaling, and, hence, can be used to represent the feature of a 2D image. The compactness in representing the shape feature with low overhead in calculation is the biggest advantage of the Moment Invariants.

In [9], the two-dimensional $(p + q)$th order moments of a density distribution function $p(x, y)$ is defined as:

$$m_{pq} = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} x^p y^q \; p(x, y)dxdy \tag{6}$$

$p, q = 0, 1, 2...$

where $p(x, y)$ is a piecewise continuous function.

It has the nonzero value only in the finite part of the x-y plane. Given the above definition, the moments of all orders exist and the uniqueness theorem can also be proved. This theorem is as follows: The double moment sequence m_{pq} is uniquely determined by $p(x, y)$; and conversely, $p(x, y)$ is uniquely determined by m_{pq}.

The characteristic function $\phi(u,v)$ and moment generating function of $p(x, y)$, $M(u, v)$ are defined as follows:

$$\phi(u,v) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} exp(iux + ivy)p(x, y)dxdy \tag{7}$$

$$M(u, v) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} exp(ux + vy)p(x, y)dxdy \tag{8}$$

If the characteristic function $\phi(u,v)$ is known, the density distribution function

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

$p(x, y)$ can be calculated by the inverse Fourier transform as follows:

$$p(x, y) = 1/(2\pi)^2 \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} exp(-iux-ivx) \ \phi(u,v)\, dudv \qquad (9)$$

The central moments are invariant under 2D translation. In [9], they are defined as $\mu_{pq} = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} (x - \bar{x})^p (y - \bar{y})^q p(x, y) d(x - \bar{x}) d(y - \bar{y})$ (10)

Where $\bar{x} = m_{10} / m_{00}$, $\bar{y} = m_{01} / m_{00}$ (11)

In [9], seven Moment Invariants are derived as follows:

$$\Phi_1 = \mu_{20} + \mu_{02},$$

$$\Phi_2 = (\mu_{20} - \mu_{02})^2 + 4\mu_{11}{}^2,$$

$$\Phi_3 = (\mu_{30} - 3\mu_{12})^2 + (3\mu_{21} - \mu_{03})^2,$$

$$\Phi_4 = (\mu_{30} + \mu_{12})^2 + (\mu_{21} + \mu_{03})^2,$$

$$\Phi_5 = (\mu_{30} - 3\mu_{12})(\mu_{30} + \mu_{12})[(\mu_{30} + \mu_{12})^2 - 3(\mu_{21} + \mu_{03})^2] + (3\mu_{21} - \mu_{03})(\mu_{21} + \mu_{03})[3(\mu_{30} + \mu_{12})^2 - (\mu_{21} + \mu_{03})^2],$$

$$\Phi_6 = (\mu_{20} - \mu_{02})[(\mu_{30} + \mu_{12})^2 - (\mu_{21} + \mu_{03})^2] + 4\mu_{11}(\mu_{30} + \mu_{12})(\mu_{21} + \mu_{03})$$

$$\Phi_7 = (3\mu_{21} - \mu_{03})(\mu_{30} + \mu_{12})[(\mu_{30} + \mu_{12})^2 - 3(\mu_{21} + \mu_{03})^2] - (\mu_{30} - 3\mu_{12})(\mu_{21} + \mu_{03})[3(\mu_{30} + \mu_{12})^2 - (\mu_{21} + \mu_{03})^2] \qquad (12)$$

2.1.2 Zernike Moments

Zernike moments are formed by Zernike polynomials proposed by Zernike in 1934 [10]. According to [10], Zernike polynomials construct a complete orthogonal basis set defined on the unit disc $(x^2 + y^2) \leq 1$. Zernike moments are defined as:

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

$$A_{mn} = \frac{m+1}{\pi} \int_x \int_y f(x, y)[V_{mn}(x, y)]^* dxdy \tag{13}$$

where $m = 0, 1, 2\ldots$ which defines the order and * denotes the complex conjugate.

The Zernike polynomial $V_{mn}(r, \theta)$ are expressed in polar coordinates as:

$$V_{mn}(r, \theta) = R_{mn}(r) exp(jn\theta) \tag{14}$$

where $V_{mn}(r, \theta)$ is defined over the unit disc $(x^2 + y^2) \leq 1$. $j = \sqrt{-1}$

and $R_{mn}(r)$ is defined as:

$$R_{mn}(r) = \sum_{S=0}^{\frac{m-|n|}{2}} (-1)^s F(m, n, s, r) \tag{15}$$

$$\text{where } F(m, n, s, r) = \frac{(m-s)!}{s!(\frac{m+|n|}{2}-s)!(\frac{m-|n|}{2}-s)!} r^{m-2s} \tag{16}$$

Constructing high order Moment Invariants is a time-consuming work while it is much easier to construct high order Zernike Moments. This is because high order Zernike Moments can be constructed by simply changing the order n and repetition m while it takes much more time to compute Moment Invariants with large values of p and q according to equation 6.

2.2 Neural Networks

Neural Networks simulate the human brain to get the power of thinking like a human. According to [11], a Neural Network is an interconnected assembly of simple processing units whose functionality is based on the human neuron. The performance of the Neural Networks depends on the inter-unit connection strengths which are called weights in practice. The simplest Neural Networks may be the TLU (Threshold Logic Unit) proposed by McCulloch and Pitts [12].

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

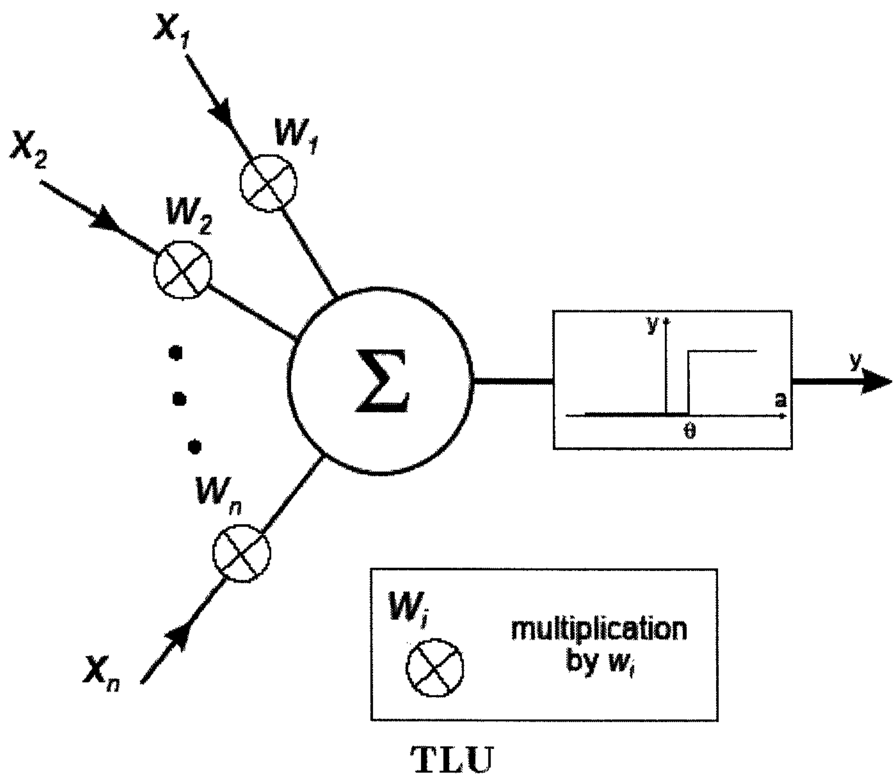


Figure 2: TLU [12]

Figure 2 shows the structure of a TLU where X_i is the input, W_i is the weight attached to X_i, θ is the threshold, y is the output, a is called the activation and is defined as:

$$a = \sum_{i=1}^{n} X_i \ W_i \tag{20}$$

If a is greater than or equal to the threshold θ, the output y will be 1, otherwise, the output y will be 0.

The Neural Networks work with two main steps, training and testing. The goal of training is to set the most appropriate weights for the specified Neural Network. This is achieved by providing input samples to the Neural Network to adjust the weights. After training, the Neural Network takes the query input and then produces output results according to the weights which have been set

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

during training step.

2.2.1 Multilayer Neural Networks

Multilayer Neural Networks are the Neural Networks which have one input layer, one output layer and one or more hidden layers as shown in Figure 3:

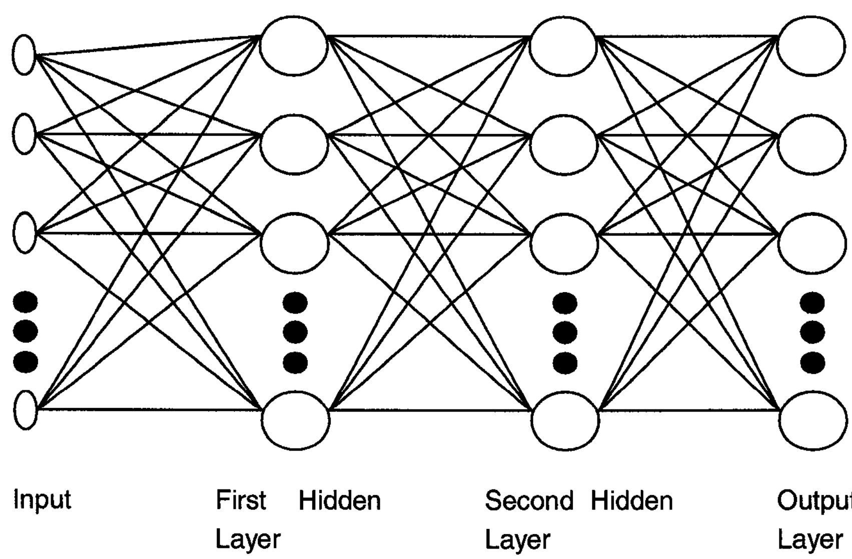


Figure 3: A Multilayer (three-layer) Neural Network

Multilayer Neural Networks have the ability to solve some more complex problems which cannot be solved by simple Neural Networks because Multilayer Neural Networks achieve more complex decision region than the simple Neural Networks. According to [13], a three-layer Neural Network can form arbitrarily complex decision regions as shown in Figure 4:

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

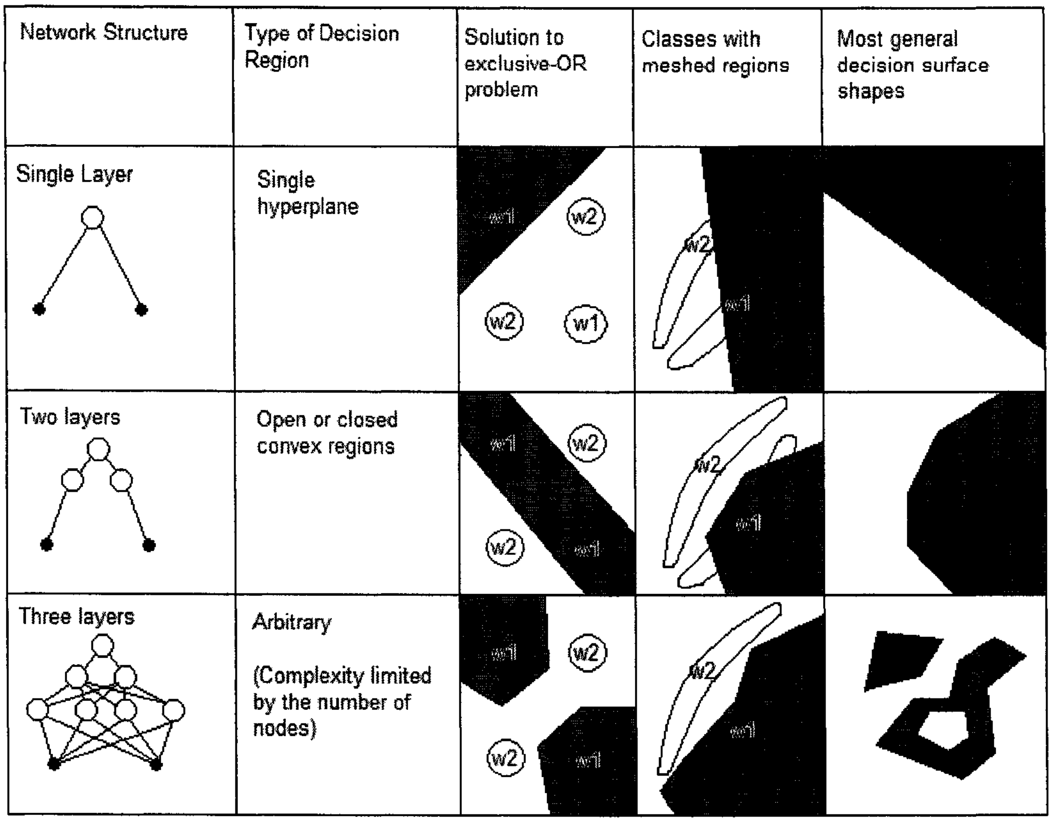


Figure 4: Types of decision regions that can be formed by single-layer and multi-layer neural networks (After: [13])

In Multilayer Neural Networks, the input data are fed to the input layer. Output of input layer becomes input to the nodes in subsequent layers, and in this way, finally reach the output layer. This process is called Feed-Forward [13]. Suppose we have a three-layer network. Let x_i denotes the input data, m, n, o denote the first hidden layer, second hidden layer and the output layer respectively, w denotes the weight, I denotes the inputs to each node of each layer. Feed-Forward process can be expressed as below:

$$I_m = \sum_i w_{mi} \; x_i \tag{21}$$

$$I_n = \sum_m w_{nm} \; A(I_m) \tag{22}$$

$$I_o = \sum_n w_{on} \; A(I_n) \tag{23}$$

where A is the activation function which is used to get continuous, soft

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

function instead of an abrupt function.

2.3 Clustering

Clustering can be considered as grouping of the 'similar' data. The goal of clustering is to make the data in the same cluster most similar and the data among different clusters most dissimilar. For example, the dots in the Figure 5 are grouped into 4 clusters.

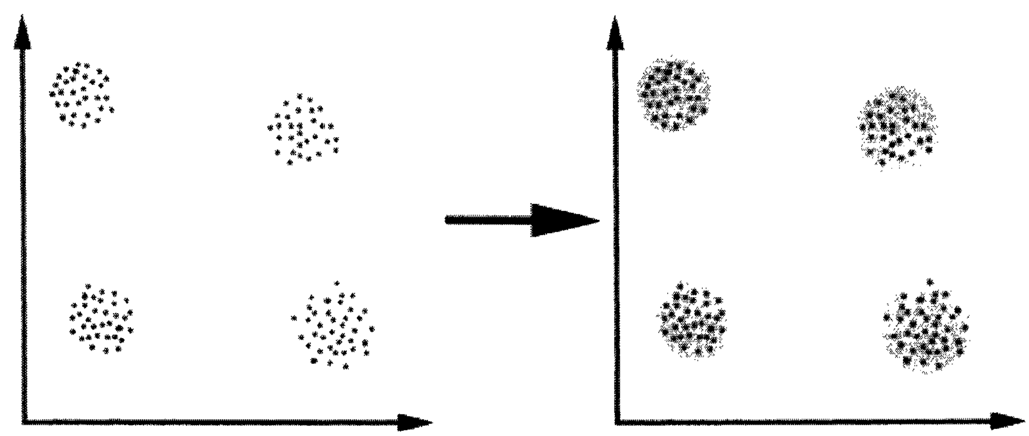


Figure 5: The dots grouped into 4 clusters [14]

Geometric distance provides the measurement of clustering. According to the goal of clustering, these dots are grouped so that the ones in the same cluster have the shortest distance between each other and the ones in the different clusters have the longest distance between each other.

According to [14], the clustering algorithm should satisfy the following requirements:

- Scalability;
- Dealing with different types of attributes;
- Discovering clusters with arbitrary shape;

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

- Minimal requirements for domain knowledge to determine input parameters;
- Ability to deal with noise and outliers;
- Insensitivity to order of input records;
- High dimensionality;
- Interpretability and usability.

The clustering technique can be classified as two main categories as follows:

(i) Exclusive clustering: the algorithm such as K-means clustering.

(ii) Overlapping clustering: the algorithm such as Fuzzy K-means clustering.

In exclusive clustering, a datum which belongs to one cluster can not belong to other clusters. On the other hand, in overlapping clustering, one datum belongs to two or more clusters with different degree of memberships.

Following sections provide some details about K-means clustering and Fuzzy K-means clustering.

2.3.1 K-Means Clustering

The K-Means Clustering involves five steps:

(1) Algorithm begins with initializing the number of clusters, say K.

(2) Then randomly put K data into K clusters such that each of these K data elements is set to be the initial centroid of the corresponding cluster.

(3) Put each of all the remaining data into the cluster which has the closest centroid.

(4) Recompute the centroids of the K clusters.

(5) Repeat Step2 to Step 4 until the centroid no longer changes.

The algorithm minimizes the squared error which is defined as follows:

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

$$J = \sum_{j=1}^{k} \sum_{i=1}^{n} ||x_i - c_j||^2 \quad (17)$$

where k is the number of clusters, n is the number of data, x_i is the ith datum, c_j is the centroid of the jth cluster.

This algorithm is said to be unsupervised because the K is set randomly. Changing the value of K affects the performance of the application.

2.3.2 Fuzzy K-Means Clustering

In real world, a datum may have some features which belong to different groups. It will be better if the datum belongs to two or more clusters. In Fuzzy K-Means Clustering, one datum belongs to one or more clusters with different degrees of membership. According to [15], it is based on the minimization of the following function:

$$J_m = \sum_{i=1}^{N} \sum_{j=1}^{C} u_{ij}{}^{m} ||x_i - c_j||^2 \quad (18)$$

where m is a real number greater than or equal to 1, u_{ij} is the degree of membership of x_i in the cluster j, x_i is the ith data, c_j is the center of the jth cluster, and $||x_i - c_j||$ is the distance from any datum to the centroid of the cluster.

The Fuzzy K-Means Clustering algorithm involves following steps:

(1) Algorithm begins with initializing the number of clusters, say K

(2) Initialize m to be a real number greater than 0

(3) Initialize the iteration counter $T = 0$

(4) Initialize $U=[u_{ij}]$ matrix, say $U^{(0)}$

(5) Initialize the stopping criterion ε (ε = 0.001 gives reasonable

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

convergence)

(6) At T^{th} step: calculate the center vectors $C^{(T)}=[c_j]$ with $U^{(T)}$ like follows:

$$c_j = \frac{\sum_{i=1}^{N} u_{ij}{}^m x_i}{\sum_{i=1}^{N} u_{ij}{}^m} \tag{19}$$

(7) Update $U^{(T+1)}$ using the formula below:

$$u_{ij} = \frac{1}{\sum_{x=1}^{k} \left(\frac{\| x_i - c_j \|}{\| x_i - c_x \|}\right)^{\frac{2}{m-1}}} \tag{20}$$

(8) If $\| U^{(T+1)} - U^{(T)} \| < \varepsilon$ then STOP; otherwise increase T by 1 and return to step 6.

Like K-Means Clustering, Fuzzy K-Means Clustering is also unsupervised because K is set randomly. Changing the value of K affects the performance of the application. In [21], a method using Simulated Annealing is proposed to determine the optimal K.

2.3.3 Distance Functions

The performance of clustering highly depends on the distance function used. Many distance functions have been proposed and used in clustering techniques. In this thesis, we investigated four distance functions, Euclidean Distance, Mahalanobis Distance, Correlation Distance, and Standardized Euclidean Distance. The details of these four distance functions are covered in Chapter 3. The performances of these distance functions are showed in Chapter 4.

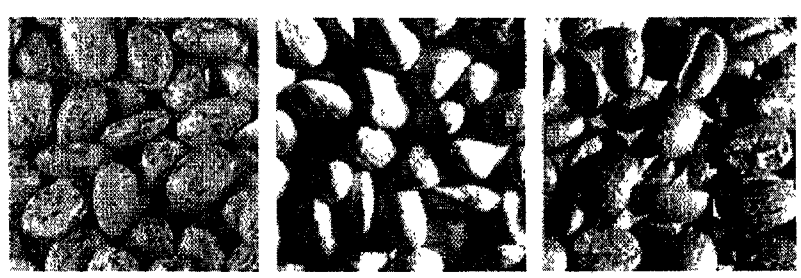
2.4 User Feedback

Although the Content-Based Image Retrieval got the most attention in past few

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

years, the User Feedback has not got enough consideration. In [16], the early method for CBIR is called Computer-Centric which is not satisfactory because of the following reasons:

- The gap between the high-level and the low-level features of the image. The high-level features refer to the objects in the image such as the ball, animals or the fruits, while low-level features are the color, shape and texture. Sometimes, the mapping from the high-level features to the low-level features is difficult for the user to do.
- The subjectivity of human perception. Different people may perceive the same image differently. One person may be more interested in the color of the image while another may be more interested in the texture. For example, in Figure 6 below, one may perceive that (a) and (b) are more similar if they do not care for the intensity contrast but the other one may think (a) and (c) are more similar because they ignore the details of the seeds.



(a) (b) (c)

Figure 6: Subjectivity of human perception [16]

Because of the above reasons, many researchers move their focus from the traditional CBIR to the interactive methods which involve User Feedback. For instance, in [16], usage of supervised learning before retrieval is introduced.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

3. Methodology

In the proposed approach, Moment Invariants and Zernike Moments are used to extract the visual features of the images. The core of our system is the Neural Network. It takes user's input--a query image and outputs the retrieved images onto the screen. These retrieved images are ranked by the similarity of the retrieved image to the query image. But before that, the Neural Network has to be trained with sample images. We could combine the Neural Network with the Clustering techniques to speed up the retrieval process. Data can be clustered in different ways. In our approach, we used Fuzzy K-means clustering technique but have also compared its performance with K-means clustering technique. We have also tested four different Distance Functions which are used in Clustering.

3.1 System architecture

The proposed shape retrieval system has 5 main stages as follows:

(1) Moment Invariants and Zernike Moments are extracted as the shape features from the images.

(2) The shape features are grouped into several clusters using Fuzzy K-means clustering.

(3) Neural Network is trained. All the shape features are taken as the inputs of the Neural Network and the cluster indices of images which have these shape features are given as the target output.

(4) Neural Network performs the testing process. User's query image is taken as the input of the Neural Network. The output of the Neural Network is used to determine the indices of the clusters which may include the relevant images.

(5) Compare the shape feature of the query image with all the images in the selected cluster and return the most similar images.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

The system architecture is shown in Figure 7.

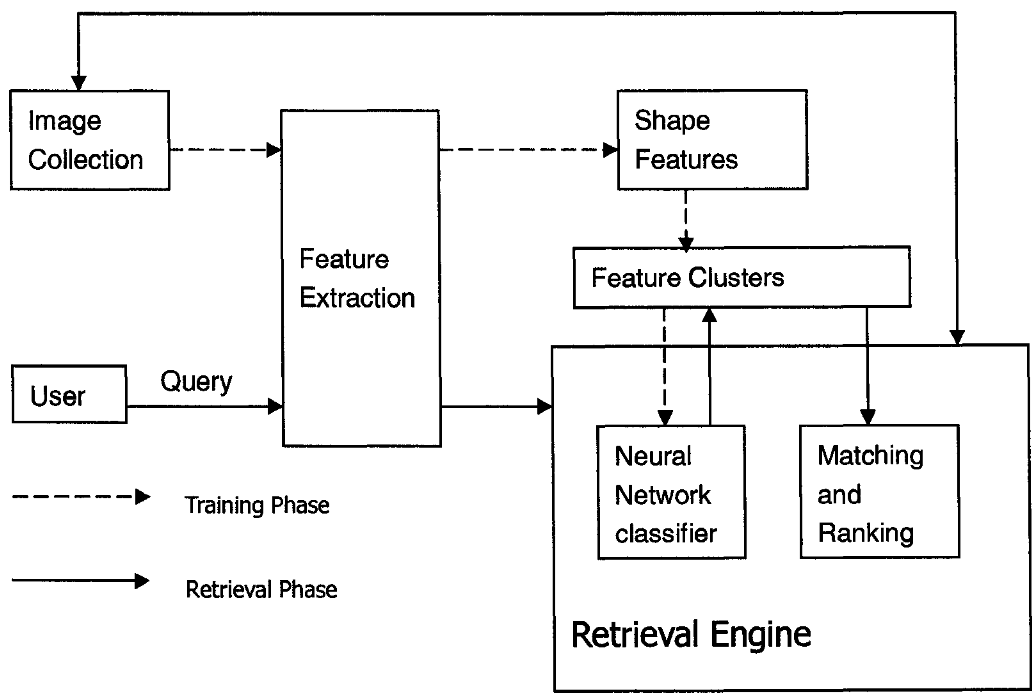


Figure 7: System Architecture

3.2 Feature Extraction

Two types of shape representations are used in this system: Moment Invariants and Zernike Moments. They both need boundary information of the image to represent the shape region. To get the boundary information, we use the Improved Turtle procedure [17]. This algorithm has the following steps [1]:

Algorithm: Finding the Boundary Sequence

Input: a digital binary image in which all pixels belonging to the shape have a value of 1 and those in background pixels have a value of 0

Output: a boundary sequence $C[i]$

Method:

(1) Find the starting point $S(s_x, s_y)$ in the image whose value is 1;

$i = 0$;

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Let the current pixel be $C[i](c_x, c_y)$, i.e., set $C = S$;

(2) Let the 4th neighbor (neighbor pixel on the left) of C be $B(b_x, b_y)$;

(3) i++

(4) Let $D[1]$, $D[2]$, ..., $D[8]$ be the 8 neighbors (anticlockwise, starting with B) of C;

Find the smallest k, $k = 1, 2, ..., 8$, that $D[k]$ has value of 1;

(5) Let $C = D[k]$ and $B = D[k\text{-}1]$;

(6) If C is the start point S then terminate;

Otherwise goto step (3).

Then, $C[i]$s are the boundary pixels, starting with S, in the anticlockwise order. After getting boundary sequence, we use it to compute Moment Invariants and Zernike Moments.

For Moment Invariants, because we are dealing with digital images, we could approximate double integrals to double summations. That means, we could approximate the $(p+q)$th moment

$$m_{pq} = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} x^p y^q \ p(x, y)dxdy$$

to

$$m_{pq} = \sum_x \sum_y x^p y^q \ p(x, y). \tag{21}$$

We can also replace $p(x, y)$ by a binary function $u(x, y)$ defined as follows to save computation time:

$u(x, y) = 1$ if (x, y) belongs to shape region, otherwise $u(x, y) = 0$. (22)

Then the equation (21) becomes

$$m_{pq} = \sum_x \sum_y x^p y^q \quad \text{where } (x, y) \text{ belongs to shape region} \tag{23}$$

We can further save the computation time by using Delta Method proposed in [18]. This method uses only the coordinates of the first pixel and the length of

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

the chained pixels of the shape region in each row to compute the moments. Then we calculate seven Moment Invariants according to Equation (12) mentioned in section 2.1.1.

For the Zernike Moments, we need to compute the centroid of the shape region first. The centroid (x_c, y_c) is computed as

$$x_c = \frac{1}{n}\sum_{i=1}^{n} x_i \, , \; y_c = \frac{1}{n}\sum_{i=1}^{n} y_i$$

where n is the number of region boundary pixels. (24)

This is followed by transformations which has the following three steps:

(1) Translation.

Translate the image origin from the top-left of the image *(u, v)* to the shape region centroid (x_c, y_c) by

$$x_t = u - x_c \, , \; y_t = v - y_c \tag{25}$$

(2) Rotation.

Flip *y* axis to make it fits the typical Cartesian coordinates by

$$x_r = x_t \, , \; y_r = -y_t \tag{26}$$

(3) Scaling.

In this final step, we have to normalize the shape region into a unit disk of Radius *R*. It is given as

$$x_s = \frac{L}{R} x_r + x_c \, , \; y_s = y_c - \frac{L}{R} y_r \tag{27}$$

We used 32 in our approach as the value of *R* to save computation time. We could also take 8, 16, or 64 because they are of the power of 2.

After transforming shape region into the unit disk, we invoke Equation (13), (14), (15), (16) mentioned in section 2.1.2 to get the six Zernike Moments.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

3.3 Fuzzy K-means Clustering

System performance highly depends on the clustering algorithm we use. If we use non-fuzzy clustering, one image could only belong to one cluster. When user queries an image, the cluster which includes this image may not be the output cluster because we cannot guarantee the images are clustered correctly.

On the other hand, if we use fuzzy clustering, the query image may belong to one or more clusters. This increases the possibility of finding the query image even though the system may get more non-relevant images.

Figure 8 shows how fuzzy clustering works. Suppose we have some data extracted from the images and we want to group them into four clusters. N_i represents image N's membership in the ith cluster. We invoke Fuzzy K-means clustering algorithm mentioned in Section 2.3.2 to update the data memberships and cluster centroids until the change is less than a stopping criterion ε (in this system, 0.001 gives reasonable result). After clustering, suppose A's memberships in four clusters are 0.1, 0.49, 0.31, and 0.1 respectively.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Cluster 1	Cluster 2	Cluster 3	Cluster 4
$A_1 = 0.1$	$A_2 = 0.49$	$A_3 = 0.31$	$A_4 = 0.1$
$B_1 = 0.2$	$B_2 = 0.35$	$B_3 = 0.35$	$B_4 = 0.1$
$C_1 = 0.5$	$C_2 = 0.1$	$C_3 = 0.05$	$C_4 = 0.35$
$D_1 = 0.3$	$D_2 = 0.35$	$D_3 = 0.2$	$D_4 = 0.15$
...	...	...	...
...	...	...	...

Figure 8: Fuzzy K-means Clustering

After clustering, data are grouped into different clusters with different memberships, but the training of the Neural Network needs the training samples to have the clear output values representing the index of the cluster they belong. That means we have to set a threshold for the membership. If datum *N*'s membership in cluster *i* is greater than this threshold, *N* belongs to cluster *i*. *N* could also belong to cluster *j* if *N*'s membership in cluster *j* is also greater than the threshold.

In Figure 8, if we set the membership threshold to 0.3, datum *A* belongs to cluster 2 and cluster 3 but not cluster 1 and cluster 2. Then we can train the neural network. After training, system is ready for retrieving images. Suppose we take *A* as the query image. In theory, system returns cluster 2 as the result cluster, but in reality, clustering is not perfect. It might return cluster 2 and cluster 1 or cluster 2 and cluster 3 or other combinations of the clusters. But whatever the return index is, *A* can be retrieved as long as it includes cluster 2 or cluster 3 because both of these clusters include *A*.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

On the other hand, if we are using non-fuzzy clustering algorithm such as K-means clustering, suppose *A* is in cluster 2, the system might not find *A* if Neural Network is not trained ideally. Although Fuzzy K-means clustering does not guarantee finding the query image, it increases the possibility of finding the query image by extending the searching range.

3.4 Neural Network

Neural Network has two main steps, training and testing. After Fuzzy K-means Clustering, training samples are ready for the training step. In our approach, we use Moment Invariants and Zernike Moments as the image feature extractors. There are seven moments for Moment Invariants and six moments for Zernike Moments. So we need two Neural Networks, one of them should have seven nodes in the input layer while the other one should have six nodes in the input layer. We choose three layers Neural Network because it can form arbitrarily complex decision region [13].

In this case, training is a two step process [19]:

- Feed-Forward: the input data feed the input layer and get each node of the next hidden layer, and in this way, finally reach the output layer. As mentioned in section 2.2.2, activation functions are used to get smooth functions instead of abrupt functions. According to [1], we choose Binary sigmoid and Bipolar sigmoid in this system because they do not need much computation time and provide good results.

  Binary sigmoid is defined as:

  $$f(x) = \frac{1}{1+e^{-x}} \tag{28}$$

  Bipolar sigmoid function is defined as:

  $$g(x) = \frac{1-e^{-x}}{1+e^{-x}} \tag{29}$$

- Back-Propagation: back-propagate the associated error of each layer

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

to its previous layer, and adjusts the weights connecting these two layers. The algorithm is as follows [1]:

Algorithm: Back-Propagation.

Input: The training samples for a multilayer feed-forward neural network.

Output: A Neural Network trained to classify the samples.

Method:

(0) Termination condition is :

(a) Minimum Squared Error (MSE) is less than a threshold ε, say $\varepsilon = 0.05$ OR

(b) |MSE(t) – MSE(t-1)| < ε and the network has gone through the training samples a maximum number of times, say 10000 times.

(1) Initialize all weights in NN

(2) while terminating condition is not satisfied {

(3) for each training sample X {

(4) // Propagate the input forward

(5) for each hidden or output layer unit j {

(6) $I_j = \sum_i w_{ij} O_i$

(7) // Compute the bet input of unit j with respect to the previous layer i

(8) $O_j = f(I_j)$; // Compute the output of each unit j

// Back-propagate the errors:

(9) for each unit j in the output layer

(10) $\delta_j = (T_j - O_j)\, f'(I_j)$; // compute the error

(11) for each unit j in the hidden layers, from the last to the first hidden layer

(12) $\delta_j = f'(I_j) \sum_k Err_k w_{jk}$;

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

// Compute the error with respect to the next higher layer k

(13) for each weight w_{ij} in NN {

$\Delta w_{ij}(t) = (\eta)\delta_j O_i + (\alpha)\Delta w_{ij}(t-1)$;

// Weight increment where $0 < \alpha, \eta < 1$ with typical values of $\eta = 0.5$ to 0.6, $\alpha = 0.8$ to 0.9

(14) $w_{ij}(t+1) = w_{ij}(t) + \Delta w_{ij}(t)$; } // Weight update

}}}

3.5 Distance Functions

Distance Function is used to measure the extent of similarity of two data items. According to [20], in a d-dimensional space, for two elements x and y, there exists a real number $D(x, y)$, called the distance function that must possess the following four properties:

(1) $D(x, y) >= 0$	(non-negativity)
(2) $D(x, y) = 0$ if and only if $x = y$	(identity)
(3) $D(x, y) = D(y, x)$	(symmetry)
(4) $D(x, y) <= D(x, z) + D(z, y)$	(triangle inequality)

In our approach, we chose and tested 4 different Distance Functions, Euclidean Distance, Correlation Distance, Standardized Euclidean Distance, and Mahalanobis Distance because they measure extent between data from different aspects. We will discuss this later in this section. First, let us see the definitions of these 4 Distance Functions.

Suppose we have two n-dimensional vectors $x = (x_1, x_2, \ldots, x_n)$ and $y = (y_1, y_2, \ldots, y_n)$, then each of these distance functions is defined as:

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

(1) Euclidean Distance is defined as:

$$dE(x, y) = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2} \tag{30}$$

(2) Correlation Distance is defined as:

$$dR(x, y) = 1 - \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2}\sqrt{\sum_{i=1}^{n}(y_i - \bar{y})^2}} \tag{31}$$

where $\bar{x} = \frac{1}{n}\sum_{i=1}^{n}x_i$, $\bar{y} = \frac{1}{n}\sum_{i=1}^{n}y_i$

(3) Standardized Euclidean Distance is defined as:

$$dSE(x, y) = \sqrt{\sum_{i=1}^{n}\frac{1}{s_i^2}(x_i - y_i)^2} \tag{32}$$

where s_i is the standard deviation of all sample vectors

(4) Mahalanobis Distance is defined as:

$$dML(x, y) = \sqrt{(x-y)^T S^{-1}(x-y)} \tag{33}$$

where S is the covariance matrix.

$$S = \frac{1}{M}\sum_{i=0}^{M-1}(x_i - u)(x_i - u)^t, \quad u = \frac{1}{M}\sum_{i=0}^{M-1}x_i$$

Where M is the number of all sample vectors.

Different Distance Functions may measure different aspects of data extent. Correlation Distance focuses on the similar variation of the feature vector as opposed to the similar numerical values which may be the focus of Euclidean Distance. Suppose we have x(1,2,3,4,5), y(100,200,300,400,500), z(5,4,3,2,1). If we are using Euclidean Distance, x and y will be placed into the same cluster whereas x and z will be placed into the same cluster if we are using Correlation Distance.

Standardized Euclidean Distance and Mahalanobis Distance could cluster data

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

with unequal coordinate variances as shown in Figure 9.

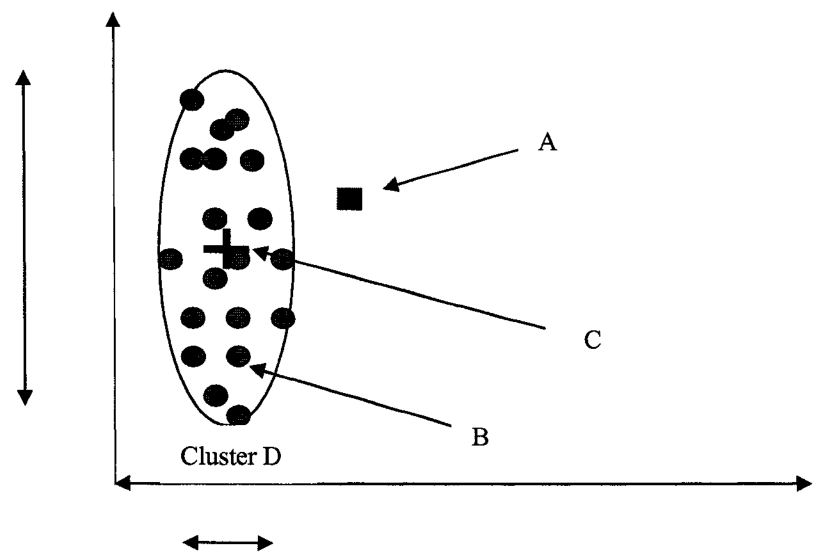


Figure 9: Data with unequal coordinate variances (After: [44])

In Figure 9, a point *A* and a point *B* have the same distance from the cluster centroid *C*. *B* belongs to the cluster *D* but *A* does not. The points outside the ellipse may not have the same distribution with the points inside. On the other hand, Euclidean Distance and Correlation Distance are good at clustering the data whose distribution is like a circle but not an ellipse as shown in Figure 10.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

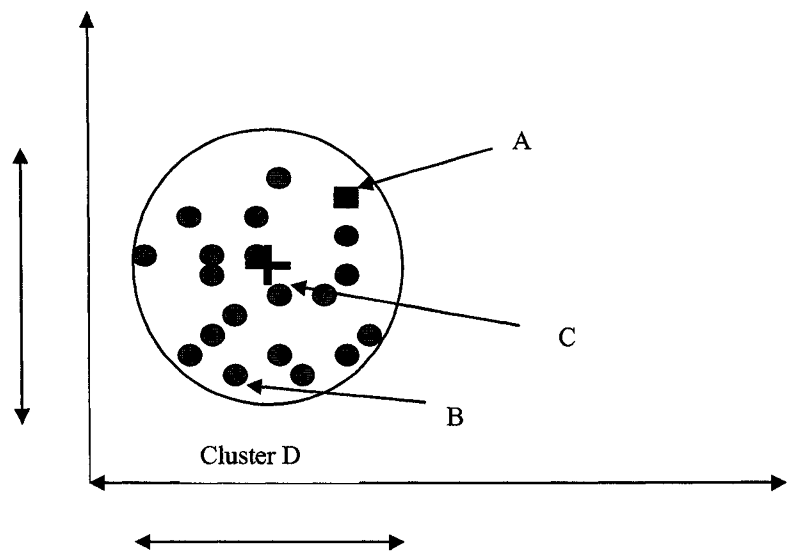


Figure 10: Data with equal coordinate variances

In Figure 10, a point *B* belongs to the cluster *D*. If a point *A* has the same distance away from the cluster centroid *C* with the point *B*, it must belong to the cluster *D*. But this is not always the case in Figure 9.

If we are dealing with feature vectors which have unequal coordinate variances, Standardized Euclidean Distance and Mahalanobis Distance will be working better than Euclidean Distance and Correlation Distance because Standardized Euclidean Distance and Mahalanobis Distance consider all elements of the feature vector equally important.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

4. Experiments and Discussions

4.1 Development Environment and GUI

We have developed our system using Visual C#.NET 2003 on a system with Windows XP running on a Dell computer which has a Pentium4M CPU of 2.6GHz and RAM of 512MB. Image features are extracted and stored in XML files. A Graphical User Interface (GUI) is provided. Users can access all functionalities such as Adding Image to the Image Collection, Clustering, Neural Network Training, and Retrieval of Images using GUI. It is showed in Figure 11.

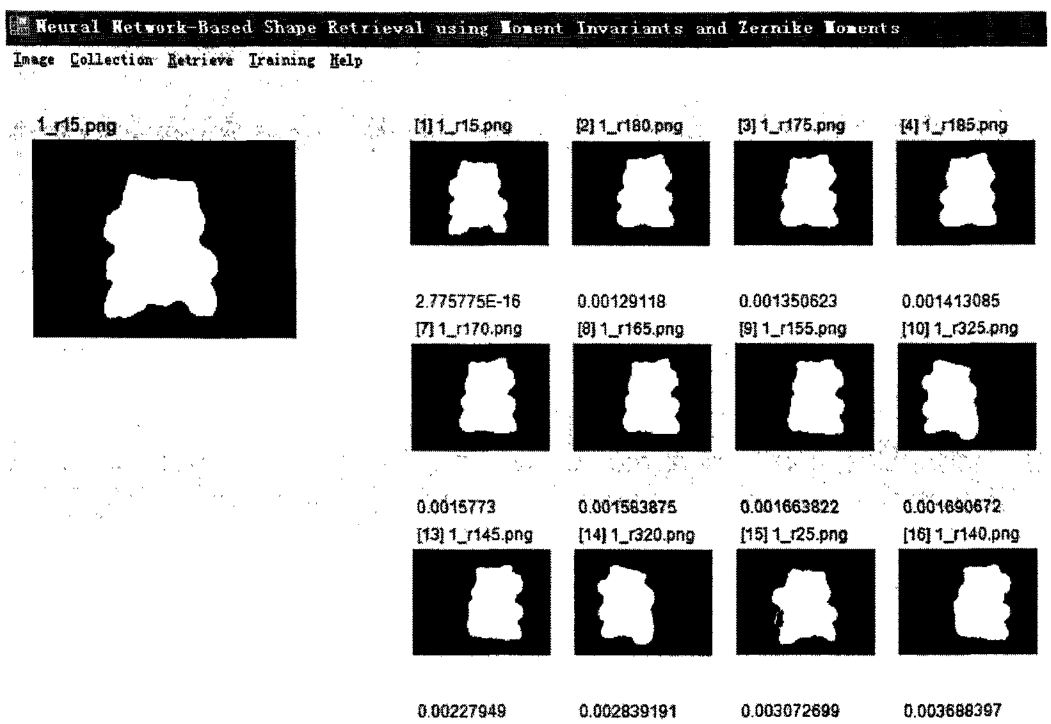


Figure 11: GUI and a sample retrieval action

As shown in Figure 11, query image is in the top left, whereas the retrieved images are shown on the right with calculated distance below them. These retrieved images are ranked by distance in ascending order.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

The training time depends on the size of the image collection and the capabilities of the machine running the system. In our case, with an image collection consisting of 10000 images, it took about 24 hours for training the Neural Networks for both Moment Invariants and Zernike Moments.

4.2 Image Collection

In order to test the system performance, we have chosen Amsterdam Library of Object Images (ALOI). ALOI uses 1000 objects to record images for scientific purposes. About 100 images have been recorded for each object. These images are recorded with different viewing angle, illumination angle, and illumination color.

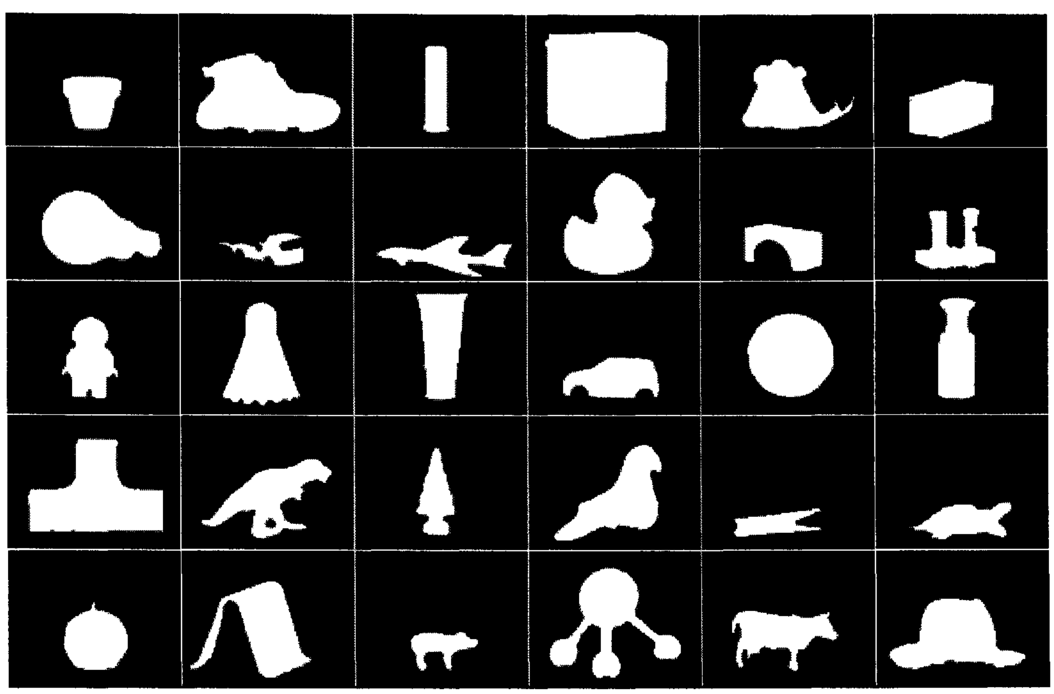


Figure 12: Sample Binary Images in the Image Collection

ALOI contains not only contain color images, but also contain binary images which are needed for our research. There are more than 100,000 binary images with different resolutions in the collection. We have selected 10,000

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

images such that each image has a resolution of 192 x 144. This selected image collection includes 200 groups of images. Each group has 50 images which are identified as relevant images, whereas images from different groups are considered as irrelevant images.

4.3 Performance Measurement

Normally, precision and recall is the most commonly used coefficient to measure the performance of the CBIR system.

In [43], Precision is defined as:

$$P = \frac{NumberOf\,\mathrm{Re}\,levant\,\mathrm{Im}\,ages\,\mathrm{Re}\,trievaed}{NumberOf\,\mathrm{Im}\,ages\,\mathrm{Re}\,trieved} \tag{34}$$

Recall is defined as:

$$R = \frac{NumberOf\,\mathrm{Re}\,levant\,\mathrm{Im}\,ages\,\mathrm{Re}\,trievaed}{TotalNumberOf\,\mathrm{Re}\,levant\,\mathrm{Im}\,agesInThe\,\mathrm{Im}\,ageCollection} \tag{35}$$

An ideal CBIR system should have 100% precision and 100% recall. But practically, it is very hard to get high precision and high recall at the same time. In some applications, we have to find as many relevant images as we can so we have to increase the recall. To do this, we often extend the range of retrieval to increase the recall, but this results in retrieval of a significant number of irrelevant images, at the expense of reduced precision. This is because that we may get much more irrelevant images than the relevant images when we extend the retrieval range.

In following sections, we present our measurement of the system performance in terms of Distance Functions, Clustering Algorithms, the number of clusters, the threshold of the Neural Network, evaluation of system performance and the training samples, determination of the number of relevant images in a cluster, and the image size.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

4.4 Comparison of Distance Functions

Unless mentioned otherwise, in all of the following experiments described in this section, the number of clusters have been chosen to be 5 with a threshold of 0.4 and a image collection consisting of 10000 images. Furthermore, *D* denotes the distance function; *Eu* denotes the Euclidean Distance; *Cor* denotes the Correlation Distance; *SE* denotes the Standardized Euclidean Distance; *Ma* denotes the Mahalanobis Distance; *Tr* denotes the threshold of the Neural Network, *K* denotes the number of clusters.

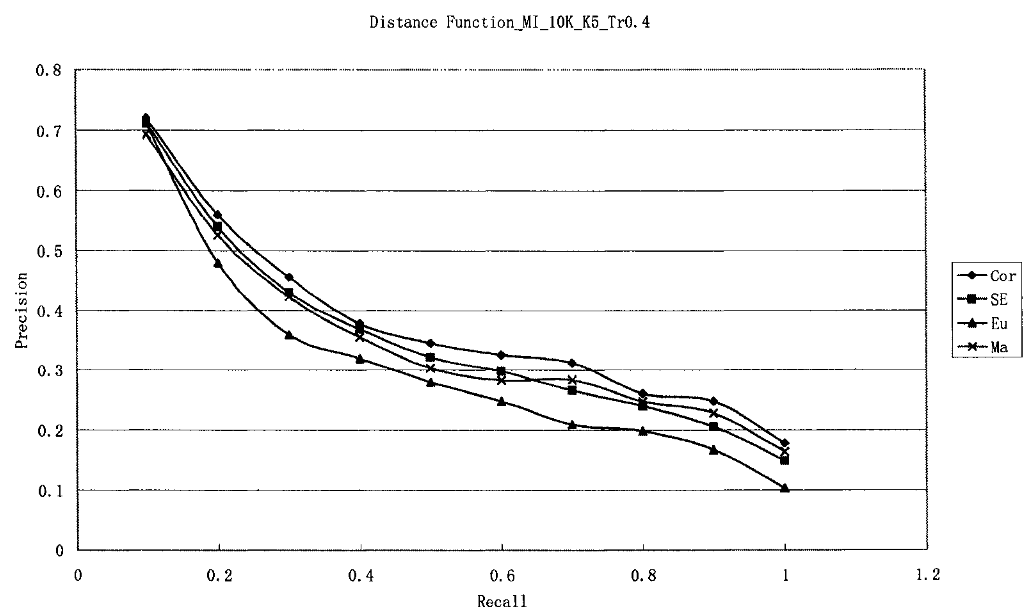


Figure 13: Comparison of Distance Function for Moment Invariants

Figure 13 shows the Comparison of four Distance Functions for Moment Invariants in precision-recall graph. For the same value of recall, a higher precision represents better performance. Generally speaking, Correlation Distance performs better than other distance functions in this case since images in a cluster have high degree of correlation for Moment Invariants.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

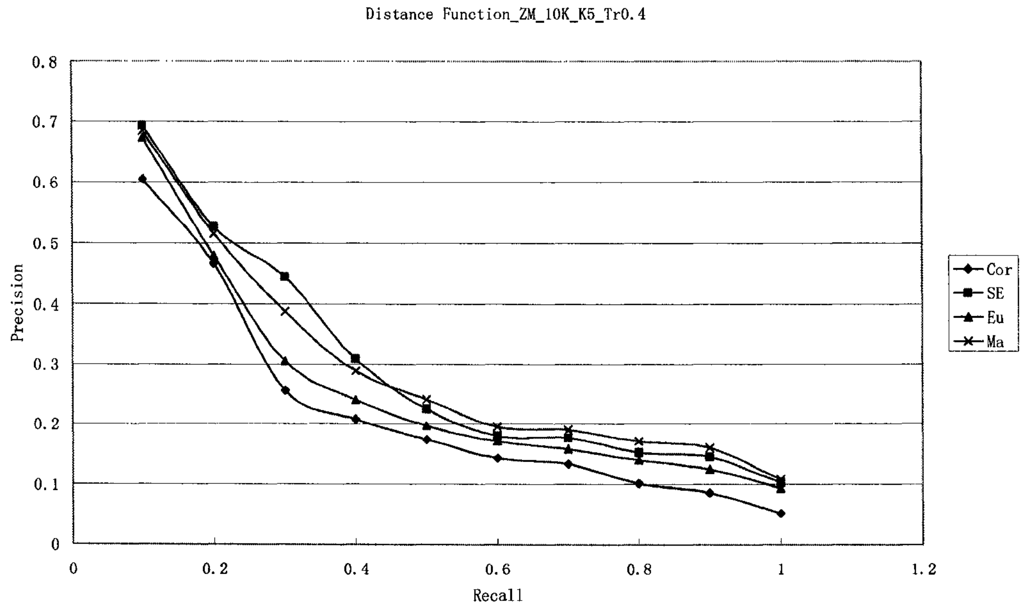


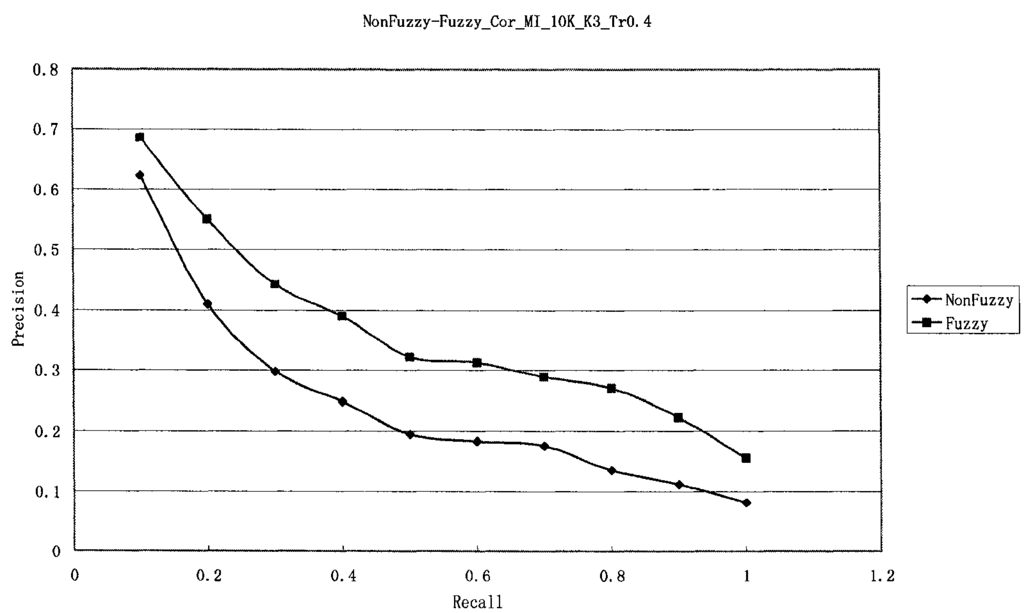
Figure 14: Comparison of Distance Function for Zernike Moments

Figure 14 shows the Comparison of four Distance Functions for Zernike Moments. In this case, Correlation Distance performs worst because of the large numerical variance of Zernike Moments. Euclidean Distance performs better than Correlation Distance because the focus of the Zernike Moments is the variance of the numerical values.

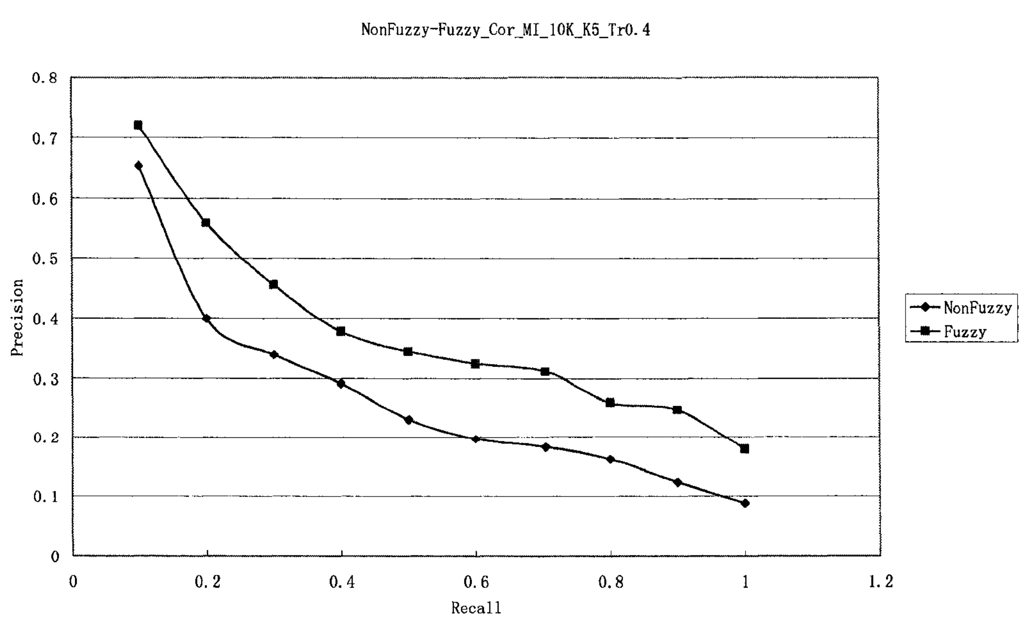
Standardized Euclidean Distance and Mahalanobis Distance perform well because they eliminate the unequal coordinate variances in the data distribution. Unequal coordinate variance is a more serious problem in Zernike Moments so Standardized Euclidean Distance and Mahalanobis Distance perform better for Zernike Moments than Moment Invariants.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

4.5 Comparison of Fuzzy K-means Clustering and K-means Clustering

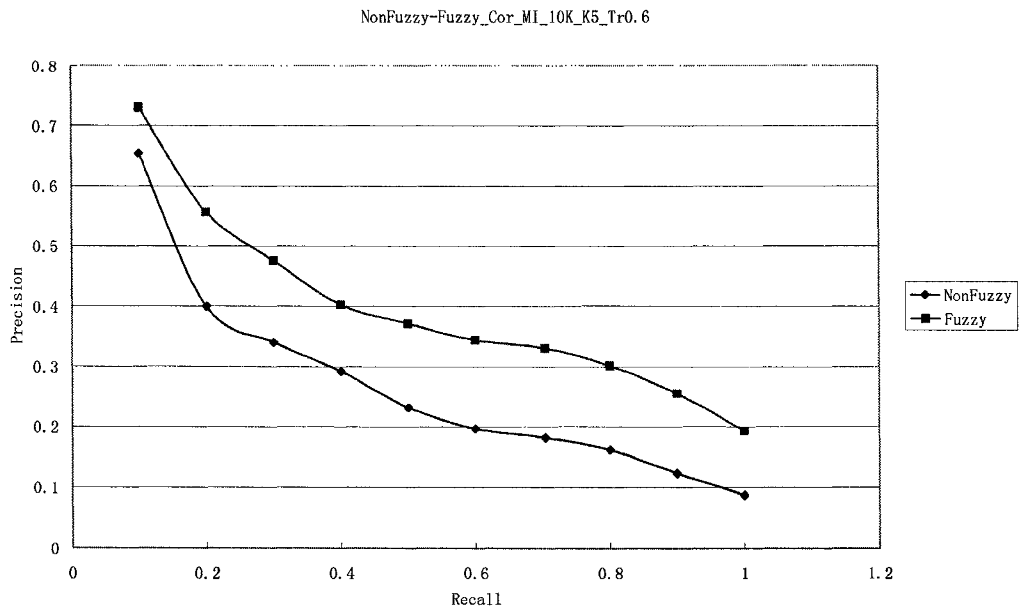


(a): *D*: *Cor*, *K*: 3; *Tr*: 0.4



(b) *D*: *Cor*, *K*: 5; *Tr*: 0.4

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

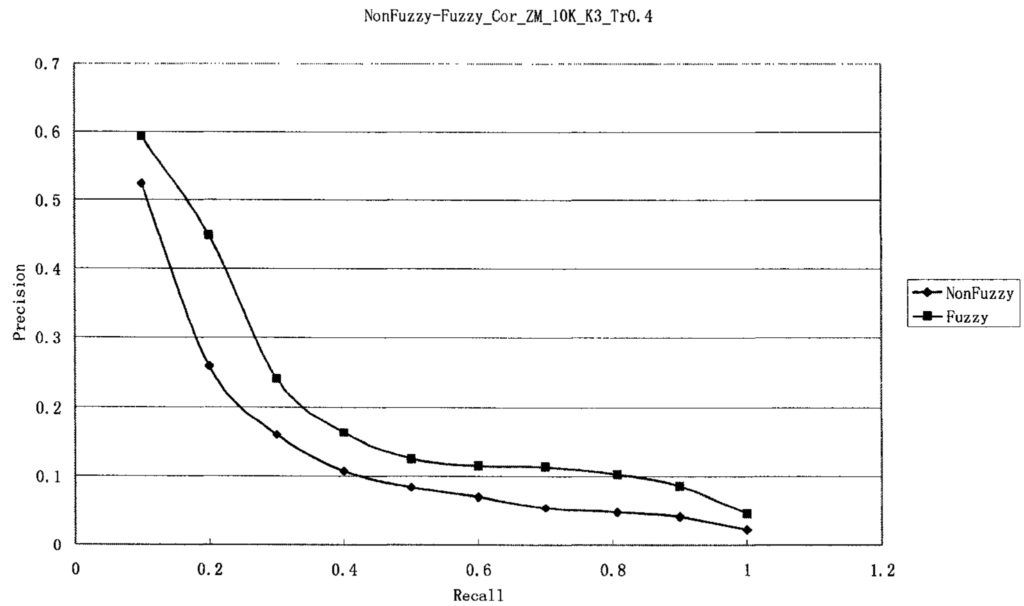


(c): *D*: *Cor*, *K*: 5; *Tr*: 0.6

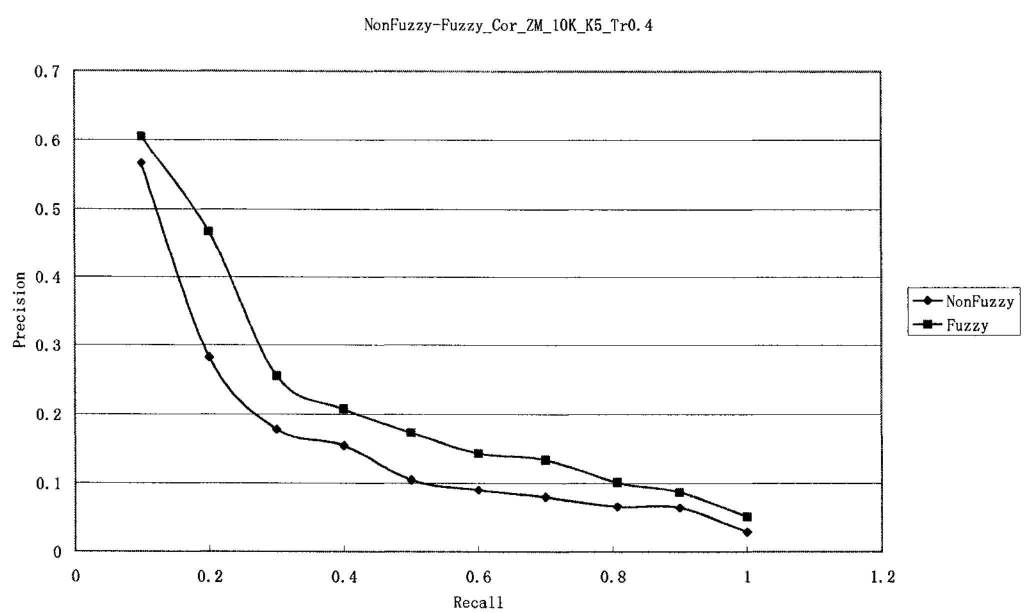
Figure 15: Comparison of Fuzzy K-means Clustering and K-means Clustering for Moment Invariants using Correlation Distance

Figure 15 shows the Comparison of Fuzzy K-means Clustering and K-means Clustering for Moment Invariants. In these three graphs, different Numbers of Clusters (3 and 5) and Neural Network Thresholds (0.4 and 0.6) have been tested. From these graphs, it is obvious that Fuzzy K-means Clustering always performs better than K-means Clustering for Moment Invariants.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

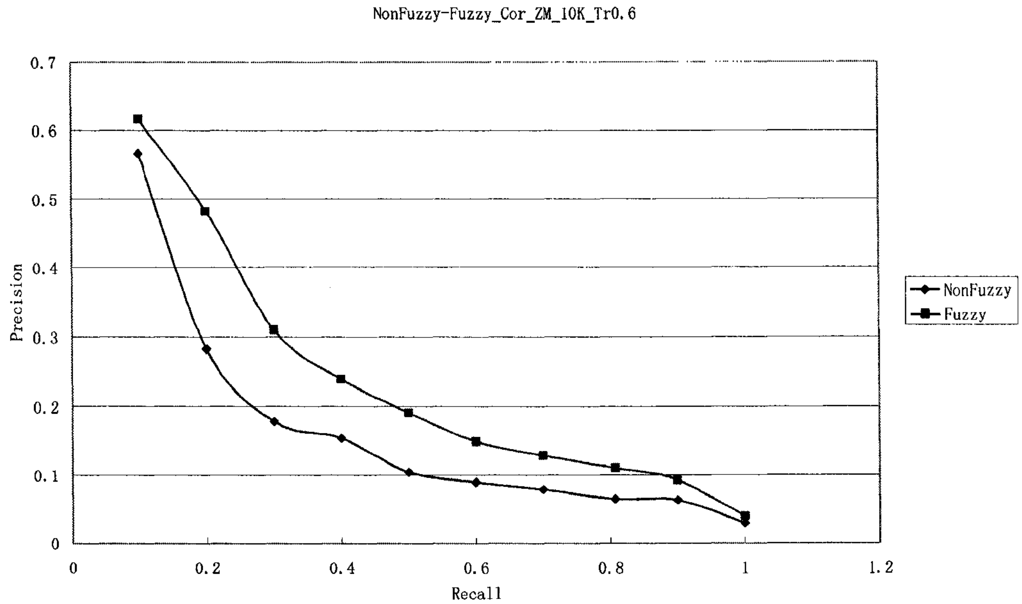


(a): *D*: *Cor*, *K*: 3; *Tr*: 0.4



(b): *D*: *Cor*, *K*: 5; *Tr*: 0.4

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

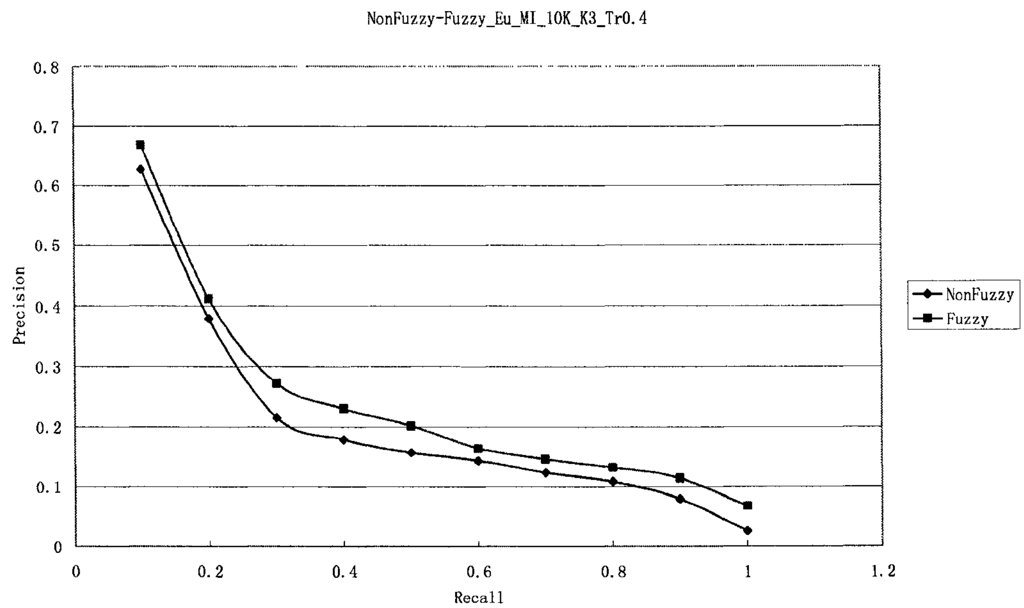


(c): *D*: *Cor*, *K*: 5; *Tr*: 0.6

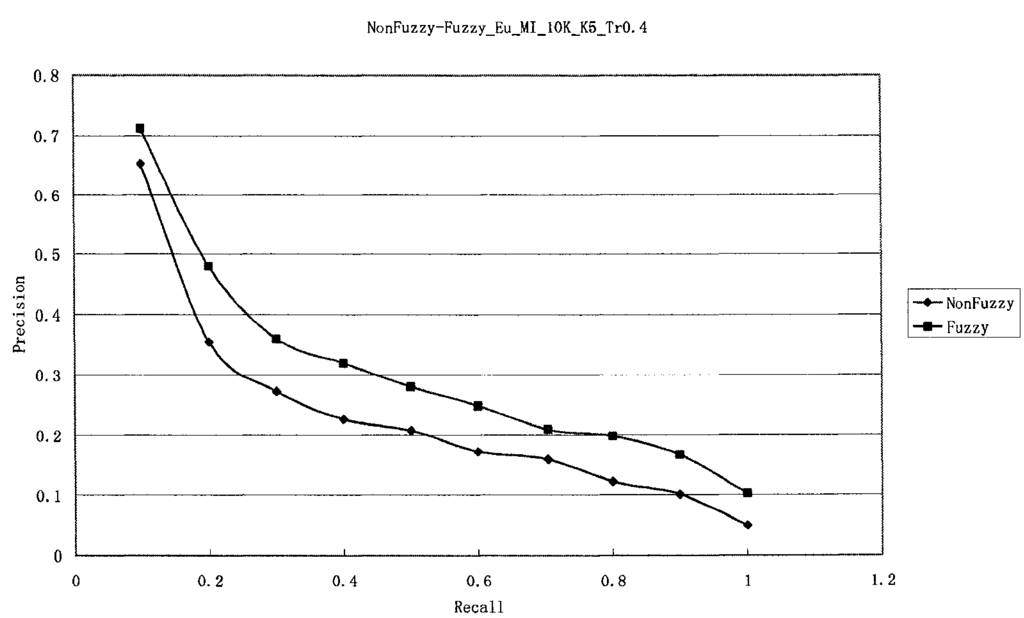
Figure 16: Comparison of Fuzzy K-means Clustering and K-means Clustering for Zernike Moments using Correlation Distance

Figure 16 above shows similar results for the Zernike Moments. Here we can also see that Fuzzy K-means Clustering is performing better than the K-means Clustering.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

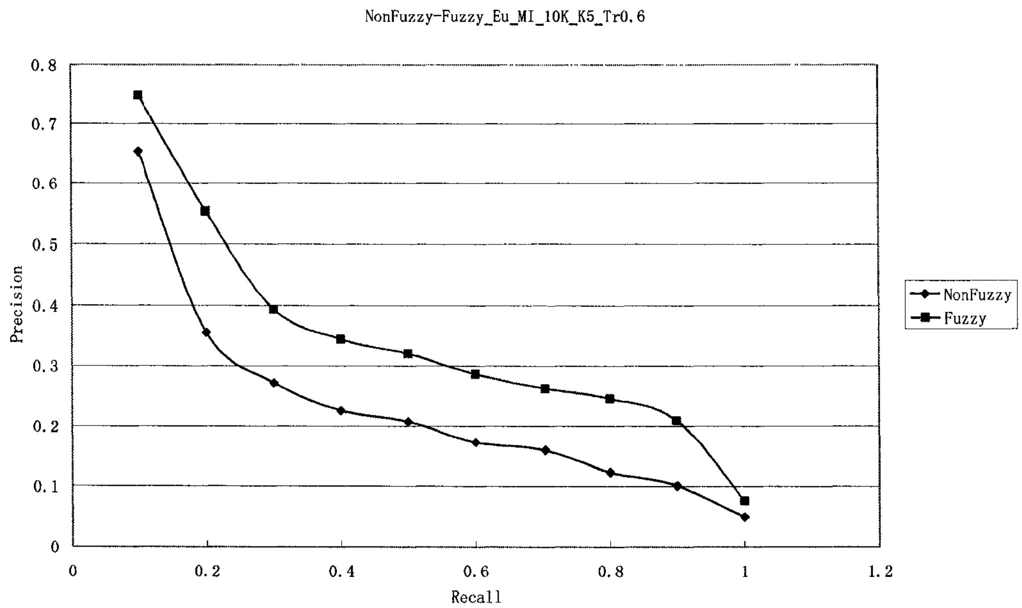


(a): *D*: *Eu*; *K*: 3; *Tr*: 0.4



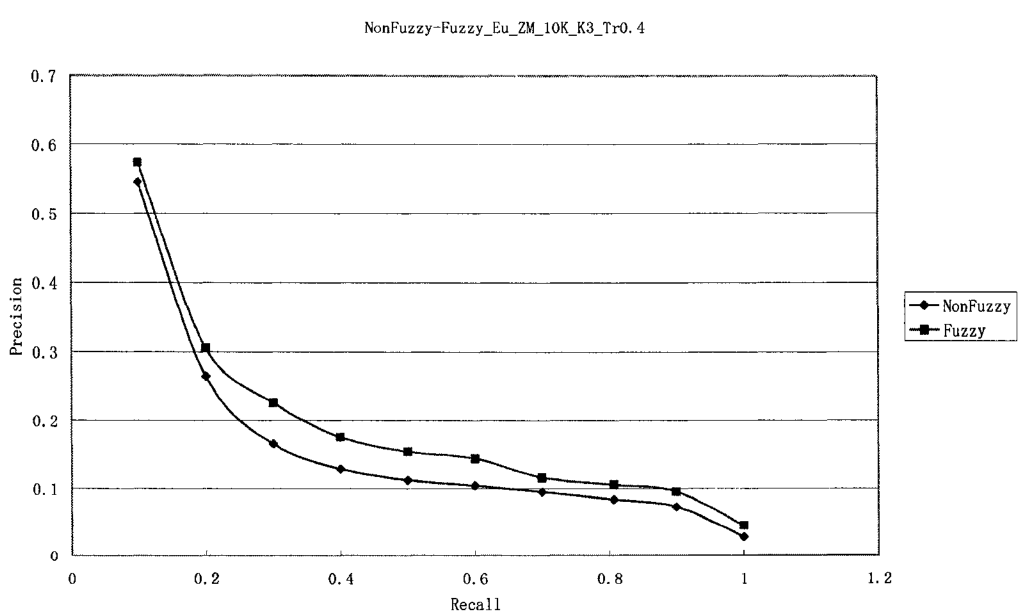
(b): *D*: *Eu*; *K*: 5; *Tr*: 0.4

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.



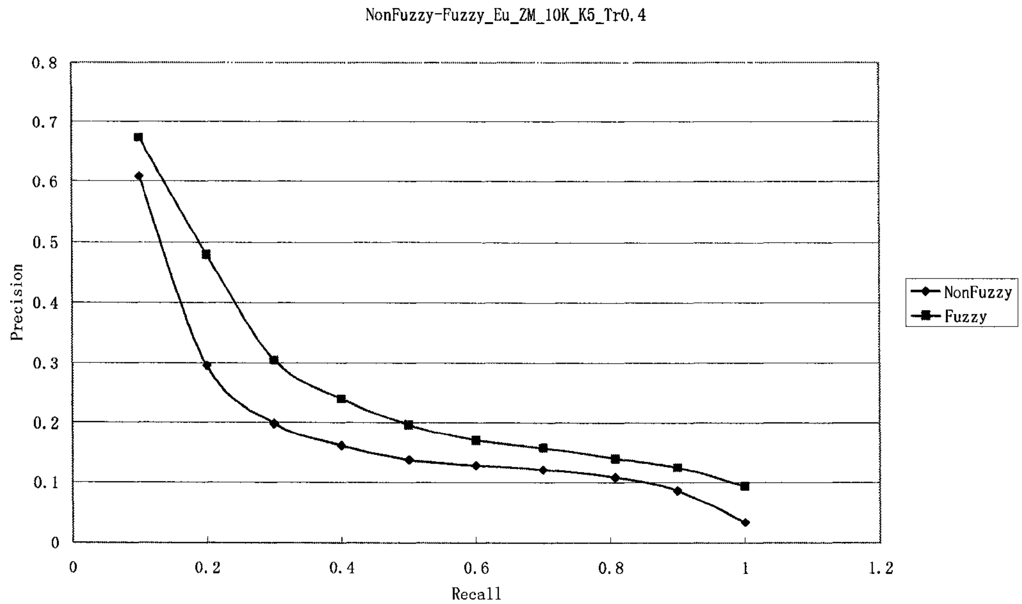
(c): *D*: *Eu*; *K*: 5; *Tr*: 0.6

Figure 17: Comparison of Fuzzy K-means Clustering and K-means Clustering for Moment Invariants using Euclidean Distance

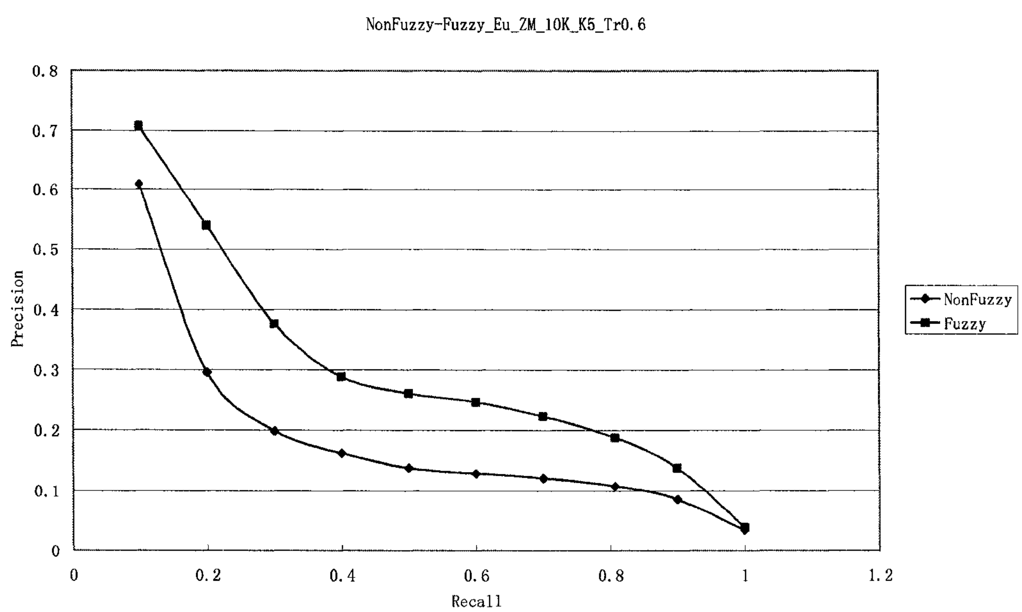


(a): *D*: *Eur*, *K*: 3; *Tr*: 0.4

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.



(b): *D*: *Eur*; *K*: 5; *Tr*: 0.4



(c): *D*: *Eu*; *K*: 5; *Tr*: 0.6

Figure 18: Comparison of Fuzzy K-means Clustering and K-means Clustering for Zernike Moments using Euclidean Distance

Figure 17, 18 provide the results of comparison between Fuzzy K-means Clustering and K-means Clustering for different number of clusters and

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

threshold values but the distance function in these experiments is Euclidean Distance. It shows that the Fuzzy K-means Clustering performs better than the K-means Clustering with Euclidean Distance as well.

4.6 Comparison of number of clusters in Fuzzy K-means Clustering

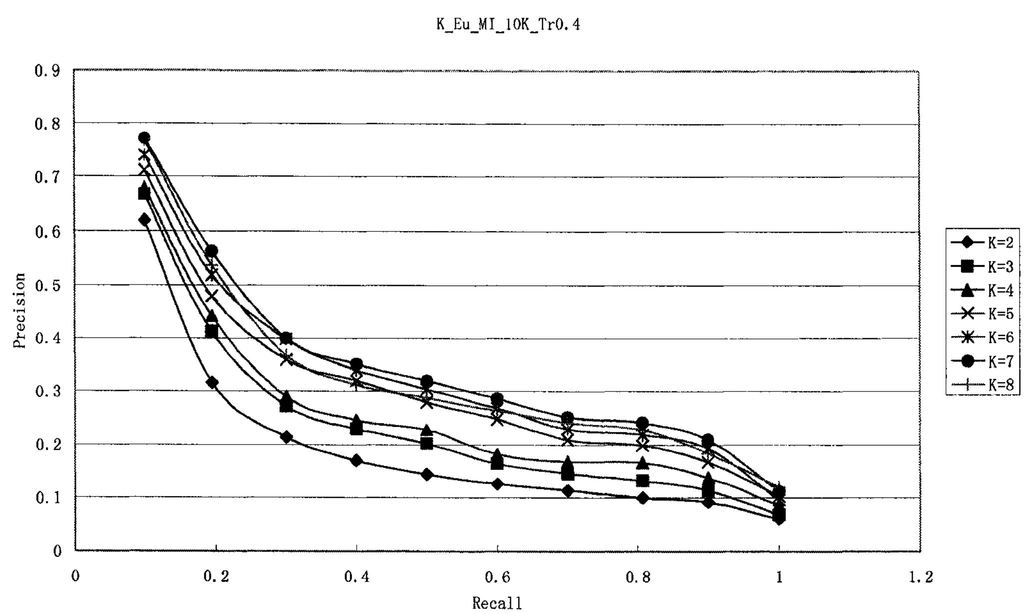


Figure 19: Comparison of Number of Clusters in Fuzzy K-means Clustering for Moment Invariants (*D*: *Eu*)

Figure 19 shows the Comparison of number of clusters for Moment Invariants. It can be observed that when $K = 7$, we get the best result. This is because of the fact that generally increasing the number of clusters reduces the number of irrelevant images in each cluster so the precision increases. But we could not guarantee that the images are always clustered correctly, and therefore, we get worse result when $K = 8$ than $K = 7$.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

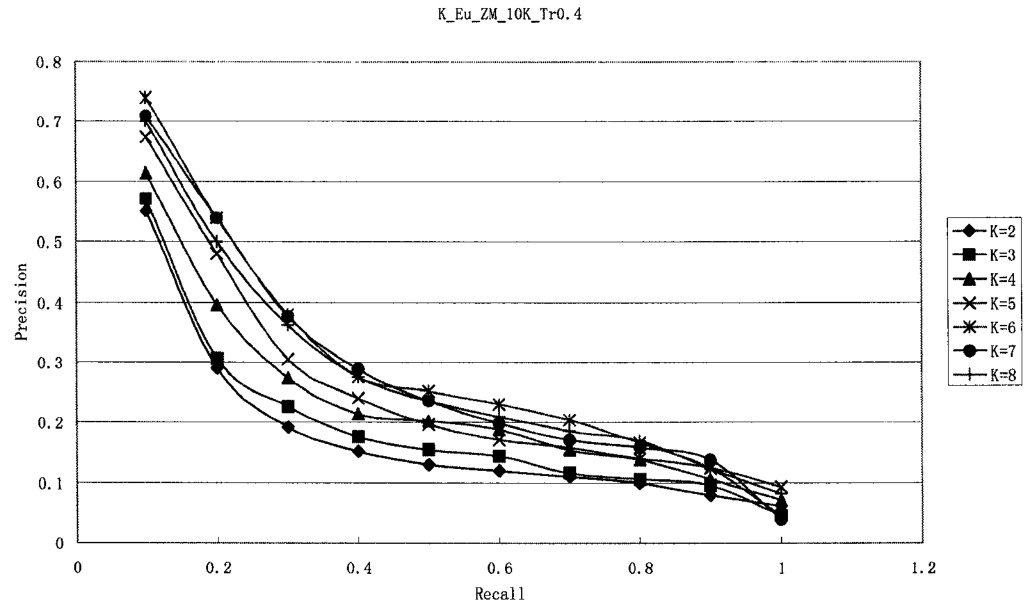


Figure 20: Comparison of Number of Clusters in Fuzzy K-means Clustering for Zernike Moments (*D*: *Eu*)

Figure 20 shows similar result for Zernike Moments. In this case, best result is achieved when $K = 6$. In [21], an approach using Simulated Annealing is introduced to find the optimal K.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

4.7 Comparison of the threshold for retrieval

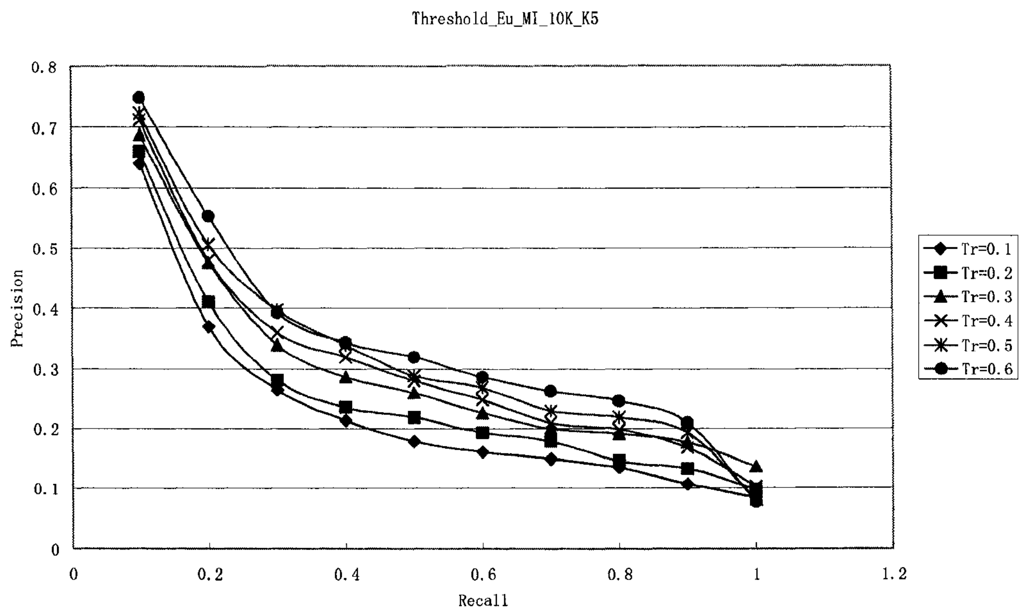


Figure 21: Comparison of the threshold for retrieval for Moment Invariants (*D*: *Eu*)

Figure 21 shows the comparison of the threshold for retrieval for Moment Invariants. Generally speaking, performance is improved by increasing the threshold. When we increase the threshold, system returns less clusters. Though we get less relevant images, we also get much less irrelevant images, so the precision is increased; when we decrease the threshold, precision is decreased because of the more irrelevant images in more returned clusters for retrieval.

We could also see that when recall reaches about 0.9, the precision of the curve which has high threshold drops dramatically. This is because the retrieval engine could not find any of the relevant images. Suppose we have 50 relevant images in an image group. First 45 images are included in the first cluster, and other 5 belong to the second cluster. When the threshold increases,

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

the second cluster is omitted by the retrieval engine. The clusters which are omitted because of the threshold always contain small amount of images. This is the reason why we may see a precision drop when the recall is very high.

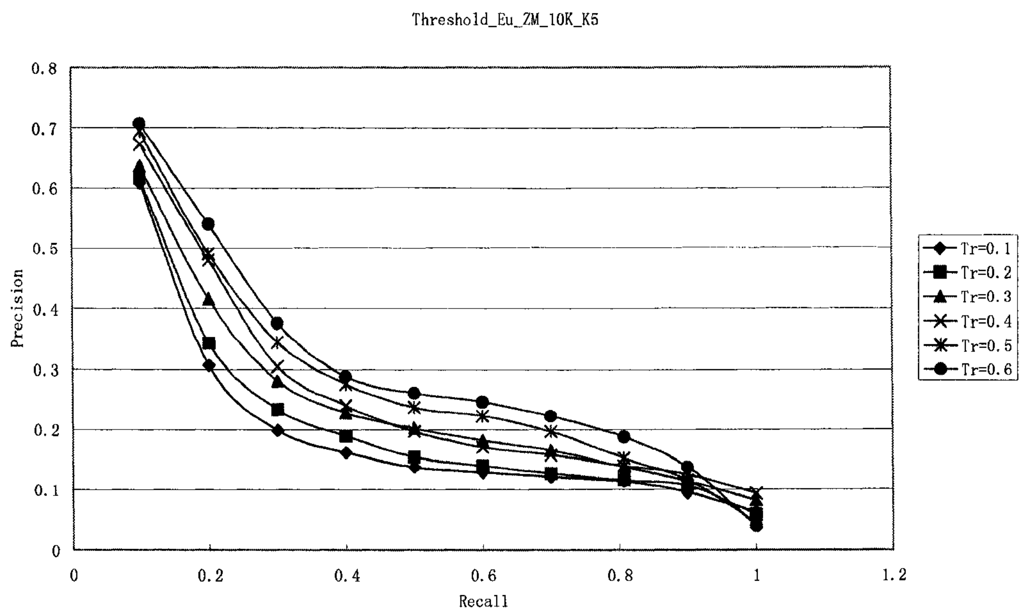


Figure 22: Comparison of the threshold for retrieval for Zernike Moments (*D*: *Eu*)

Figure 22 shows the comparison of the threshold for retrieval for Zernike Moments. The results are similar to those given in Figure 21. Though high threshold gives us high precision, we can't use the threshold as high as 0.9 or even 1.0. It will let the retrieval engine skip too many clusters so that we may not get any relevant images.

Selecting a good threshold value for optimal results is application dependent. For instance, the retrieval of medical images usually requires the system to find as many images as it can, so we can't set the threshold too high, whereas if we just want to find a place to visit from some landscapes, we could set a higher threshold to eliminate more irrelevant places we do not want to go.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

4.8 Evaluation of System Performance and the Training Samples

We have tested our system for cases, when the images used for the system training involved query image, as well as experiments in which the query images were not included in the images used for the system training.

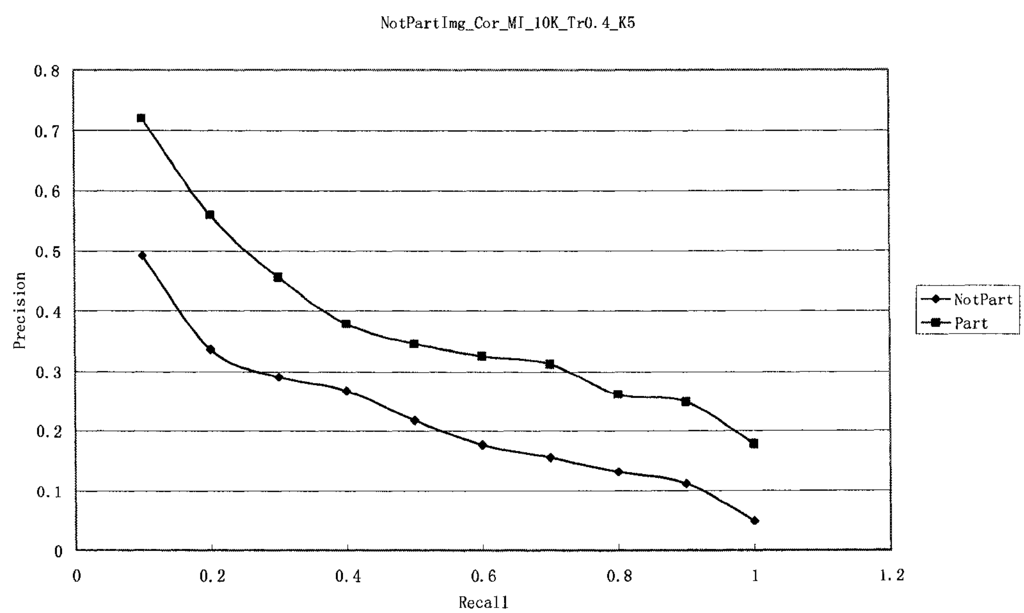


Figure 23: Evaluation of System Performance and the Training Samples for Moment Invariants (*D*: *Cor*)

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

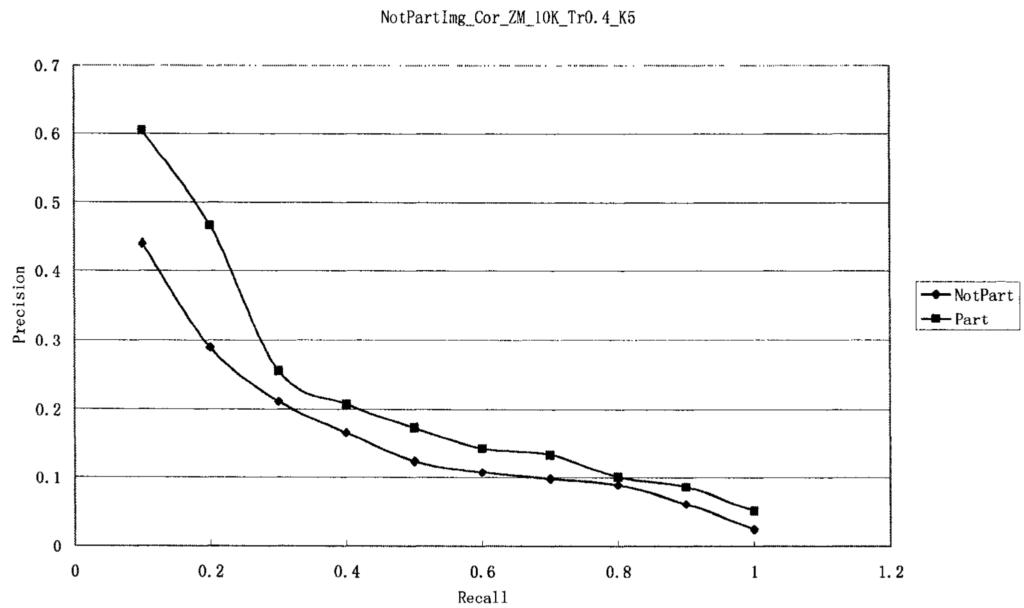


Figure 24: Evaluation of System Performance and the Training Samples for Zernike Moments (*D*: *Cor*)

As we can see in Figure 23 and 24, precision is worse if the query image does not belong to the training samples. That is because the weights of the Neural Network is set for retrieving the training samples, not the images not belong to the training samples. The errors of the weights because of the unfamiliar input will affect the performance of the Neural Network.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

4.9 Determination of the number of relevant images in a cluster

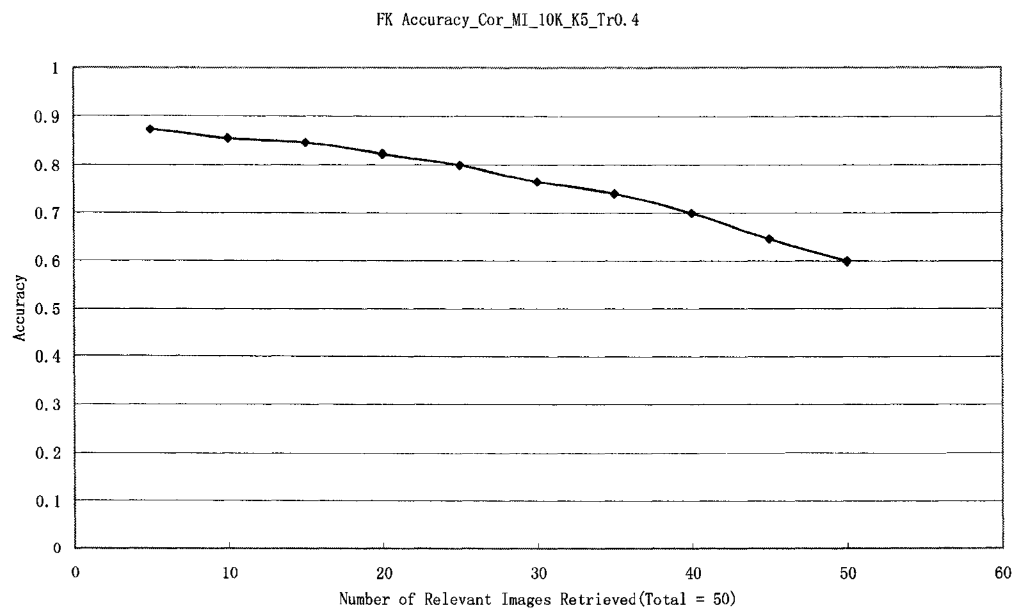
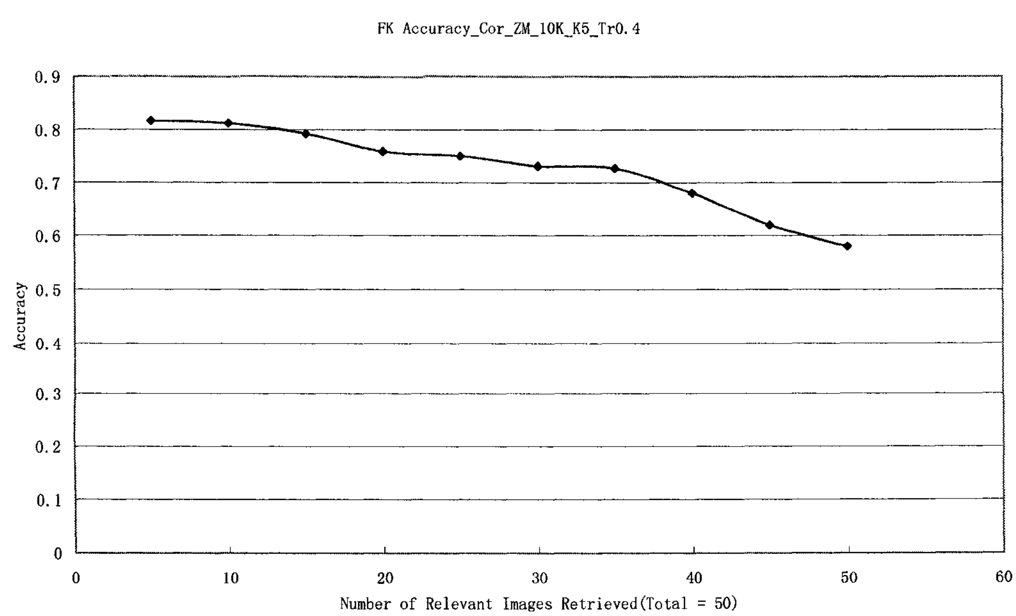


Figure 25: Determination of the number of relevant images in a cluster for Moment Invariants (*D*: *Cor*)



Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Figure 26 Determination of the number of relevant images in a cluster for Zernike Moments (*D: Cor*)

Figure 25 and 26 show the Fuzzy K-means Clustering accuracy. Our image collection contains 200 groups of images; each of them has 50 relevant images.

If we are using Moment Invariants, about 88% of the first 5 images (ranked by similarity) belong to the first cluster the Neural Network returns. About 60% of all of the 50 images belong to the cluster that the Neural Network has determined to be the most relevant.

If we are using Zernike Moments, about 82% of the first 5 images (ranked by similarity) belong to the first cluster the Neural Network returns. About 58% of all of the 50 images belong to the cluster that the Neural Network has determined to be the most relevant.

Because of the high accuracy of the Fuzzy K-means Clustering, if we do not have to find all the relevant images, we could stop retrieval when we get enough relevant images we need to save computation time.

4.10 Comparison of Image Sizes

For the experiment, we have used the image collection containing 200 images for the comparison of the images sizes. As mentioned in Chapter 2, geometrically transforming the image should not affect the retrieval result because Moment Invariants and Zernike Moments are invariant to them.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

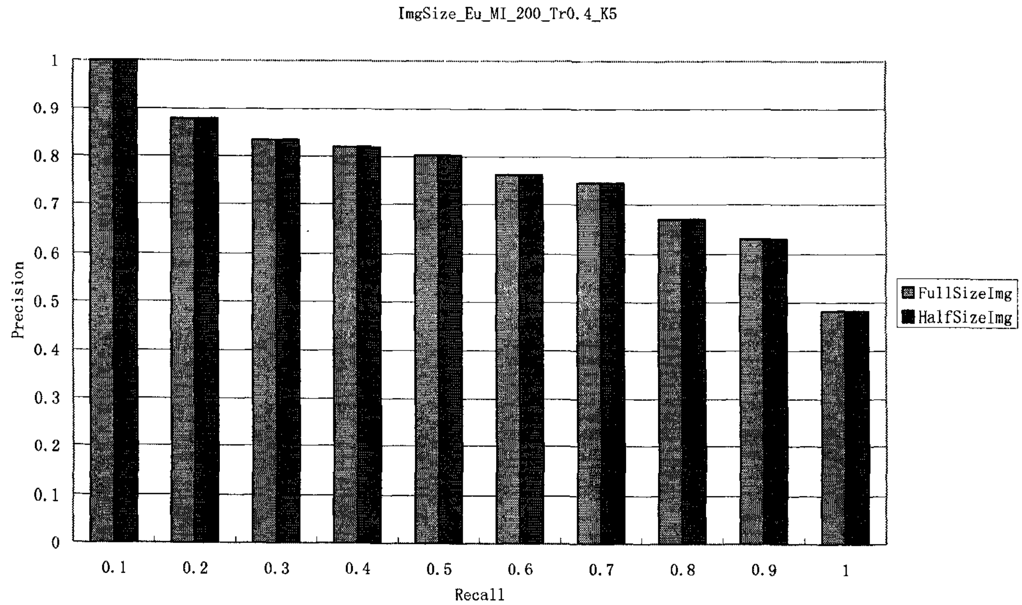


Figure 27: Comparison of Image Sizes for Moment Invariants (*D*: *Eu*; Number of Images of the Image Collection: 200)

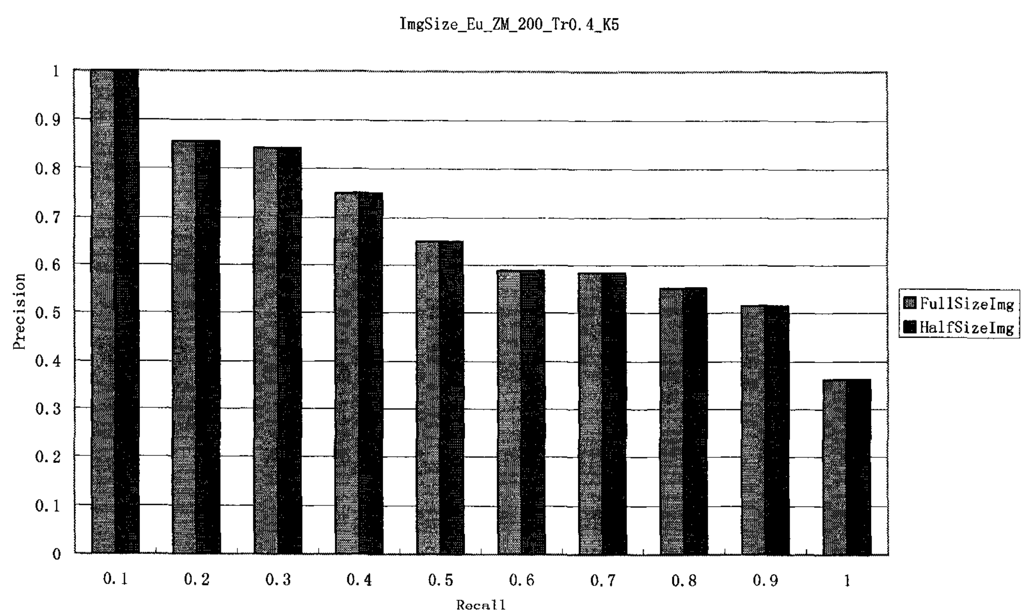


Figure 28: Comparison of Image Sizes for Zernike Moments (*D*: *Eu*; Number of Images of the Image Collection: 200)

In Figure 27 and Figure 28, we used images with different pixel dimensions

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

(sizes) of 192 x 144 and 96 x 72. It indicates that the results do not change regardless of the size of images.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

5 Conclusion and Future Works

5.1 Conclusion

Use of Clustering to group correlated image and Neural Network as a classifier and retrieval engine make it possible to perform efficient image retrieval even in large image database.

The main goal of this thesis is to implement and evaluate Fuzzy K-means Clustering using Neural Network and various distance functions and compare its performance against an existing system given in [1]. Results show that Fuzzy K-means Clustering performs better than K-means Clustering.

We implemented and compared Euclidean, Correlation, Standardized Euclidean and Mahalanobis distance functions in our system. Results show we could choose different distance functions with different applications to get better performance.

5.2 Future Directions

Our approach can be improved in following ways:

1. Improving Clustering by finding the optimal number of clusters.

   System performance may be improved by finding the optimal number of clusters. In [21], a method using Simulated Annealing is proposed to determine the optimal K.

2. Adding User Feedback

   Different people may perceive the same picture differently, this will provide a supplementary information to the retrieval engine to improve the performance.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

References

[1] X. Chen, I. S. Ahmad. "Neural Network-Based Shape Retrieval Using Moment Invariants". In Frontiers of Information Technology, pp. 131-138, Islamabad, Pakistan, Dec 20-21, 2006.

[2] I. S. Ahmad. "Image Indexing and Retrieval Using Moment Invariants". In Proceedings of the Fourth International Workshop on Information Integration and Web-based Applications and Services (iiWAS), pp. 93-104, Bandung, Indonesia, Sep 10-12, 2002.

[3] Y. Rui, TS Huang, S. Chang. "Image retrieval: Past, present, and future". In Journal of Visual Communication and Image Representation, Vol. 10, pp. 1-23, 1999.

[4] M. Stricker, M. Orengo. "Similarity of color images". In Proceedings of Storage and Retrieval for Image and Video Databases (SPIE), Vol. 2420, pp. 381-392, San Jose, CA, USA, Feb 1995.

[5] J.R. Smith, Shih-Fu Chang. "Single color extraction and image query". In Proceedings of International Conference on Image Processing (ICIP), Vol. 3, pp. 528-531, Washington, DC, USA, Oct 1995.

[6] J.R. Smith, Shih-Fu Chang "Automated binary texture feature sets for image retrieval". In Proceedings of International Conference on Acoustics, Speech, and Signal Processing (ICASSP), Vol. 4, pp. 2239-2242, Atlanta, GA, USA, May 1996.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

[7] J.R. Smith, Shih-Fu Chang. "Transform features for texture classification and discrimination in large image databases". In Proceedings of International Conference on Image Processing Image (ICIP), Vol. 3, pp. 407-411, Austin, TX, USA, Nov 1994.

[8] Sven Loncaric. "A survey of shape analysis techniques". In Pattern Recognition, Vol. 31, pp. 983-1001, Aug 1998.

[9] M.K. Hu. "Visual Pattern Recognition by Moment Invariants". In IEEE Transactions on Information Theory, Vol. 50, pp. 179-187, 1962.

[10] Jamie Shutler. "Complex Zernike Moments". URL: http://homepages.inf.ed.ac.uk/rbf/CVonline/LOCAL_COPIES/SHUTLER3/node11.html, Apr 2006.

[11] Kevin Gurney. "Neural Nets". URL: http://www.shef.ac.uk/psychology/gurney/notes/index.html, Apr 2006.

[12] W.S. McCulloch, W.H. Pitts. "A logical calculus of the ideas immanent in nervous activity". In Neurocomputing: Foundations of Research, pp. 15-27, MIT Press, 1988.

[13] R.P. Lippmann. "An Introduction to Computing with Neural Nets". In ACM SIGARCH Computer Architecture News, Vol. 16, pp. 7-25, ACM Press, 1987.

[14] Matteo Matteucci. "A Tutorial on Clustering Algorithms". URL: http://www.elet.polimi.it/upload/matteucc/Clustering/tutorial_html/, May 2006.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

[15] Australian Centre for Precision Agriculture. "FuzME: Fuzzy k-Means with Extragrades Program". http://www.usyd.edu.au/su/agric/acpa/fkme/FkME.html, May 2006.

[16] Y. Rui, S. Huang, M. Ortega, S. Mehrotra. "Relevance Feedback: A Power Tool for Interactive Content-Based Image Retrieval". In IEEE Transactions on Circuits and Systems for Video Technology, Vol. 8, pp. 644-655, Sep 1998.

[17] I. Pitas. "Digital Image Processing Algorithms". Prentice Hall, 1993.

[18] M. F. Zakaria, L. J. Vroomen, P. J. A. Zsombor-Murray, J. M. H. M.Vankessel. "Fast Algorithm for the Computation of Moment Invariants". In Pattern Recognition. Vol. 20, pp. 639-643, 1987.

[19] L. Fausett. "Fundamentals of Neural Networks—Architectures, Algorithms, and Applications". Prentice Hall, 1994.

[20] R. C. Gonzalez, R. E. Woods. "Digital Image Processing". Prentice Hall, 2002.

[21] Wei Yang, Luis Rueda, Alioune Ngom. "A Simulated Annealing Approach to Find the Optimal Parameters for Fuzzy Clustering Microarray Data". In Proceedings of the 25th International Conference on Chilean Computer Science Society, pp. 45-54, Washington, DC, USA, Nov 2005.

[22] R.C. Veltkamp, M. Tanase. "Content-based image retrieval systems: a survey". URL: http://www.aa-lab.cs.uu.nl/cbirsurvey/cbir-survey/, Feb 2006.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

[23]R. Datta, Jia Li, J. Z. Wang. "Content-based Image Retrieval – Approaches and Trends of the New Age". In Proceedings of the 7th ACM SIGMM international Workshop on Multimedia information Retrieval, pp. 253-262, Hilton, Singapore, Nov 10-11, 2005.

[24]Sagarmay Deb, Yanchun Zhang. "An overview of content-based image retrieval techniques". In Proceedings of the 18th International Conference on Advanced Information Networking and Applications, Vol. 2, pp. 59-64, Fukuoka, Japan, Mar 2004.

[25]J.R. Smith. "Image retrieval evaluation". In Proceedings of IEEE Workshop on Content-Based Access of Image and Video Libraries, pp. 112-113, Santa Barbara, CA, USA, Jun 1998.

[26]R. Krishnapuram, S. Medasani, Sung-Hwan Jung, Young-Sik Choi, R. Balasubramaniam. "Content-based image retrieval based on a fuzzy approach". In IEEE Transactions on Knowledge and Data Engineering, Vol. 16, pp. 1185-1199, Oct 2004.

[27]J. Favela, V. Meza. "Image-retrieval agent: integrating image content and text". In IEEE Transactions on Intelligent Systems and Their Applications, Vol. 14, pp. 36-39, Sep 1999.

[28]R. Tello. "Fourier descriptors for computer graphics". In IEEE Transactions on Systems, Man and Cybernetics, Vol. 25, pp. 861-865, May 1995.

[29]T. McElroy, E. Wilson, G. Anspach. "Fourier descriptors and neural networks for shape classification". In Proceedings of International Conference on Acoustics, Speech, and Signal Processing (ICASSP), Vol. 5, pp. 3435-3438, Detroit, MI, USA, May 1995.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

[30]Qing Chen, Emil Petriu, Xiaoli Yang. “A comparative study of Fourier descriptors and Hu's seven moment invariants for image recognition”. In Proceedings of Canadian Conference on Electrical and Computer Engineering (CCECE), Vol. 1, pp. 103-106, Niagara Fall, ON, Canada, May 2004.

[31]L. Kotoulas, I. Andreadis. “Real-Time Computation of Zernike Moments”. In IEEE Transactions on Circuits and System for Video Technology, Vol. 15, pp. 801-809, June 2005.

[32]Wei Yang. “Optimizing Parameters in Fuzzy K-Means for Clustering Microarray Data”. M.Sc. Dissertation, School of Computer Sciences, University of Windsor, Windsor, ON, Canada, Jul 2005.

[33]N. Watanabe, T. Imaizumi. “Fuzzy k-means clustering with crisp regions”. In Proceedings of the 10th IEEE International Conference on Fuzzy Systems, Vol. 1, pp. 199-202, Melbourne, Australia, Dec 2001.

[34]M. Sato, Y. Sato. “Fuzzy clustering model for fuzzy data”. In Proceedings of IEEE International Conference on Fuzzy Systems, Vol. 4, pp. 2123-2128, Yokohama, Japan, Mar 1995.

[35]S. Miyamoto. “An overview and new methods in fuzzy clustering”. In Proceedings of Second International Conference on Knowledge-Based Intelligent Electronic Systems (KES), Vol. 1, pp. 33-40, Adelaide, Australia Apr 1998.

[36]M. Sato, Y. Sato, L.C. Jain. “General fuzzy clustering model and neural networks”. In Proceedings of Electronic Technology Directions to the Year

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

2000, pp. 104-112, Adelaide, Australia, May 2005.

[37] A. Baraldi, P. Blonda. "A survey of fuzzy clustering algorithms for pattern recognition. II". In IEEE Transactions on Systems, Man and Cybernetics, Vol. 29, pp. 786-801, Dec 1999.

[38] Min Yao, Weihong Wang. "A kind of practical fuzzy clustering". In Proceedings of the 4th World Congress on Intelligent Control and Automation, Vol. 3, pp. 2140-2143, Shanghai, China, June 2002.

[39] Ming-Chuan Hung, Don-Lin Yang. "An efficient Fuzzy C-Means clustering algorithm". In Proceedings of IEEE International Conference on Data Mining, pp. 225-232, San Jose, CA, USA, Nov 2001.

[40] Yi Shen, Hong Shi, Jian Qiu Zhang. "Improvement and optimization of a fuzzy C-means clustering algorithm". In Proceedings of the 18th IEEE Instrumentation and Measurement Technology Conference, Vol. 3, pp. 1430-1433, Budapest, Hungary, May 2001.

[41] H. Cardot, O. Lezoray. "Graph of neural networks for pattern recognition". In Proceedings of the 16th International Conference on Pattern Recognition, Vol. 2, pp. 873-876, Quebec, Canada, Aug 2002.

[42] K. Mahesh. "Text Retrieval Quality: A Primer". URL: http://www.oracle.com/technology/products/text/htdocs/imt_quality.htm, Jul 2006.

[43] S. Draghici. "Data Analysis Tools For DNA Microarrays". CRC Press, 2003.

[44] J. R. Smith, S. F. Chang. "Tools and Techniques for Color Image Retrieval".

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

In Proceedings of Storage and Retrieval for Image and Video Databases, Vol. 2670, pp. 426-437, 1996.

[45] P. D. Wasserman. “Neural Computing: Theory and Practice”. Van Nostrand Reinhold, 1989.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

VITA AUCTORIS

NAME:	NAN XING
PLACE OF BIRTH:	Beijing, China
YEAR OF BIRTH:	1981
EDUCATION:	Beijing Institute of Technology, Beijing 1999-2003 B. Sc
	University of Windsor, Windsor, Ontario 2004-2007 M. Sc

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.